LoAD: Low-Rank Distributional Alignment and Test-Time Decision Adaptation for Zero-Shot Skeleton Action Recognition

Anonymous authors
Paper under double-blind review

Abstract

Generative methods for zero-shot skeleton action recognition mostly follow an align-then-classify paradigm, where a cross-modal autoencoder aligns skeleton sequences with class semantics in a shared latent space, upon which a classifier is built. This paper revisits both phases. On the alignment side, class-conditional latent distributions are typically modeled as diagonal Gaussians, leaving the correlation structure within each distribution unexploited. On the decision side, different decisions usually read the same generative score, whereas they in fact impose different requirements on the evidence. Accordingly, we propose LoAD, which improves this paradigm in both alignment depth and decision structure. First, Low-Rank Covariance Alignment (LRCA) extends the latent distribution to a diagonal-plus-low-rank structure, and jointly aligns the means, variances, and low-rank correlation factors of both modalities during training. Second, Test-time Decision Adaptation (TDA) refines how decisions are made at test time, with two components. Decision-Decoupled Inference (DDI) constructs two complementary per-class evidences and reads them in a decoupled manner, where class ranking in the zero-shot setting fuses a flow-matching velocity error with a deterministic reconstruction error, while seen/unseen discrimination in the generalized setting relies on the former only. Test-time Shrinkage Calibration (TSC) further adapts the class latents with statistics from unlabeled test batches, modulated by the number of seen classes and requiring no gradient updates. On standard splits of NTU-60, NTU-120, and PKU-MMD, LoAD achieves competitive performance under both zero-shot and generalized zero-shot settings, and systematic diagnostic studies verify the effectiveness of each design.

1 Introduction

Skeleton-based action recognition has benefited from powerful graph convolutional backbones Yan et al. (2018); Shi et al. (2019); Cheng et al. (2020); Chen et al. (2021), but models trained this way only recognize the classes covered by the training annotations, and annotating every new action class is expensive. Zero-shot action recognition (ZSL) removes this dependency by describing every class with a semantic embedding from a pretrained text encoder Radford et al. (2021), so that a model trained on the seen classes recognizes unseen ones from their descriptions alone Frome et al. (2013); Akata et al. (2015); Wray et al. (2019); Gupta et al. (2021); Pourpanah et al. (2023). The generalized setting (GZSL) further requires telling whether a test sample comes from a seen or an unseen class, where models are biased toward the classes they were trained on Chao et al. (2016); Xian et al. (2019a).

The dominant generative approach follows an align-then-classify paradigm Xian et al. (2019b;c); Schönfeld et al. (2019); Chen et al. (2026b). A cross-modal autoencoder aligns skeleton sequences and class semantics in a shared latent space with diagonal-Gaussian class conditionals Kingma & Welling (2014), and test-time decisions score each candidate class by how well its latent explains the test encoding under the flow-matching framework Lipman et al. (2023) (Figure 1, top). This design faces two difficulties in the zero-shot setting. The conditionals of unseen classes are predicted from semantics alone and receive no correction from skeleton observations, so their drift from where the

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

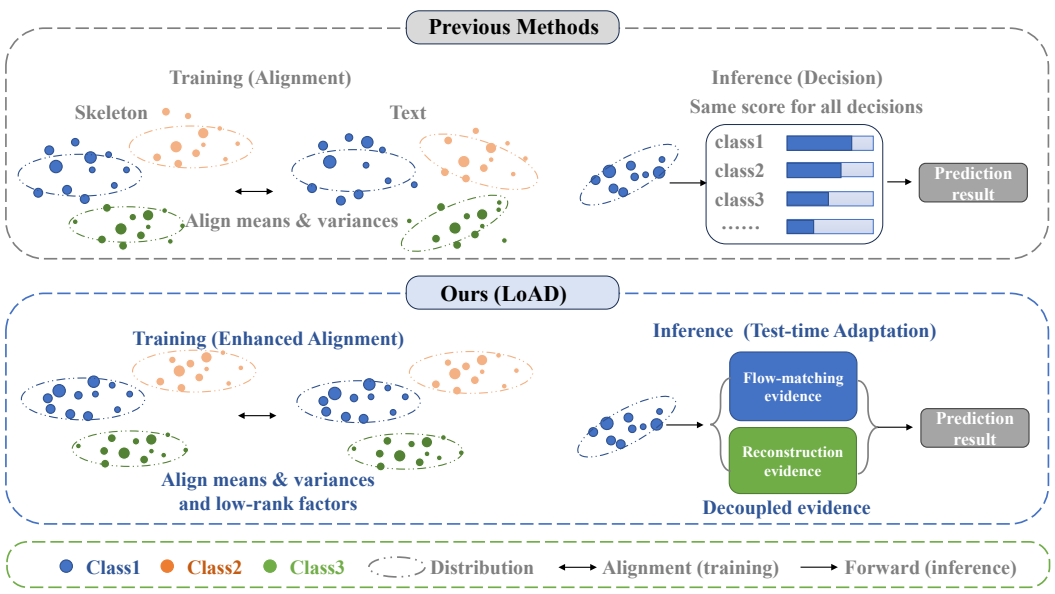


Figure 1: **Comparison with existing methods.** Existing methods align skeleton and semantic latent distributions by matching their means and variances and apply the same score to different test-time decisions (top). LoAD addresses both limitations: LRCA additionally aligns low-rank covariance factors during training, while TDA assigns distinct roles to flow-matching and reconstruction scores in ZSL ranking and GZSL gating (bottom).

class data actually lies passes directly into recognition errors. Meanwhile, ranking unseen classes in ZSL and separating seen from unseen in GZSL place different demands on the evidence, yet both rely on a single score estimated at one flow time from one latent draw and thus inherit its sampling noise.

To address these limitations, we propose LoAD (Figure 1, bottom), which revises both distribution alignment and test-time decision making. During training, Low-Rank Covariance Alignment (LRCA, Sec. 3.3) extends each class conditional to a diagonal-plus-low-rank form Ledoit & Wolf (2004) and aligns its means, variances, and low-rank factors across modalities on seen classes. Its near-zero initialization keeps the model close to the diagonal baseline at the start of training. At inference, Test-Time Decision Adaptation (TDA, Sec. 3.4) modifies the decision rule while keeping the trained model fixed, unlike test-time adaptation approaches that update model parameters, internal statistics, or class prototypes Sun et al. (2020); Wang et al. (2021); Iwasawa & Matsuo (2021); Zhu et al. (2026). Specifically, Decision-Decoupled Inference (DDI) combines a multi-time flow-matching score with a deterministic reconstruction score for ZSL ranking but retains only the former for GZSL domain gating. Drawing on the transductive use of unlabeled test data Fu et al. (2015); Liang et al. (2020), Test-Time Shrinkage Calibration (TSC) further re-anchors class latents using test-batch statistics, with its activation determined by the number of seen classes.

Together, these components target the two limitations identified above: restricted distribution modeling and coupled test-time scoring. We summarize our main contributions as follows:

- **Low-rank distribution alignment.** LRCA enriches diagonal latent covariances with compact low-rank components and aligns their statistics across skeleton and semantic modalities.
- **Decision-specific test-time inference.** TDA keeps the trained network fixed while assigning distinct roles to flow-matching and reconstruction scores in ZSL ranking and GZSL gating, with TSC providing calibration from unlabeled test-batch statistics.
- **Comprehensive empirical evaluation.** Experiments on three benchmarks show consistent ZSL gains over the direct baseline and competitive GZSL performance. Ablations and diagnostics further identify when each component is effective.

2 Related Work

2.1 Zero-Shot Skeleton Action Recognition

Zero-shot skeleton action recognition aims to recognize action classes unseen during training with the help of semantic descriptions Wray et al. (2019); Gupta et al. (2021); Pourpanah et al. (2023). Early embedding-based methods transfer knowledge from seen to unseen classes through the semantic space, by composing pose subsemblies or synthesizing class-specific features Wray et al. (2019); Gupta et al. (2021), and later works strengthen the two modalities by maximizing skeleton-text mutual information Zhou et al. (2023), learning part-aware unified representations Zhu et al. (2024), aligning the modalities in both directions Kuang et al. (2026), or modeling evolving context Chen et al. (2025); Wang et al. (2026). Generative methods learn a shared latent space with cross-modal autoencoders Schönfeld et al. (2019) and score candidate semantics against the test encoding, with variants that disentangle the latent space Li et al. (2024) or introduce frequency-semantic cues Wu et al. (2025), while recent work reads flow-matching errors as the score Lipman et al. (2023); Chen et al. (2026b). These efforts mostly enrich the semantic descriptions, the encoders, or the generative objectives, while LoAD revisits the distribution form of the class conditionals and the way the evidence is read at test time.

2.2 Test-Time Adaptation

Test-time adaptation updates a trained model on unlabeled test data, through self-supervised auxiliary tasks Sun et al. (2020), entropy minimization Wang et al. (2021), or classifier adjustment with test statistics Iwasawa & Matsuo (2021), and source-free transfer shapes target clusters with pseudo labels Liang et al. (2020). In zero-shot learning, exploiting unlabeled test data has a long transductive history Fu et al. (2015), and in skeleton-based ZSL, DynaPURLS optimizes dynamic prototypes by gradient descent at test time Zhu et al. (2026). These methods update parameters, statistics, or prototypes through backpropagation, whereas TDA adapts only the decision of a frozen model, requiring no gradient updates, no retraining, and no additional parameters.

3 Method

3.1 Preliminaries

Problem setup. Let $\boldsymbol{x} \in \mathbb{R}^{C\times T\times V}$ denote a skeleton sequence with C coordinate channels, T frames, and V joints, and let $\mathcal{S}$ and $\mathcal{U}$ denote the disjoint sets of seen and unseen classes. Every class $c \in \mathcal{S} \cup \mathcal{U}$ is described by a semantic embedding $\boldsymbol{a}_c$ from a pretrained text encoder. Training uses only skeleton sequences of seen classes; we write $\mathcal{X}_c$ for the training samples of class c. At test time, the zero-shot (ZSL) setting evaluates top-1 accuracy on unseen classes, while the generalized setting (GZSL) additionally classifies seen-class test samples and reports the seen accuracy, the unseen accuracy, and their harmonic mean, where the main difficulty is the inherent bias of GZSL models toward the classes they were trained on.

Cross-modal flow matching. We adopt the cross-modal flow-matching framework of Chen et al. (2026b) as our baseline. A skeleton backbone f maps $\boldsymbol{x}$ to an embedding $\boldsymbol{e} = f(\boldsymbol{x}) \in \mathbb{R}^{d_e}$, and two modality encoders, E^s for skeletons and E^t for semantics, take $\boldsymbol{e}$ and $\boldsymbol{a}_c$ as input and output the parameters of diagonal Gaussian conditionals in the shared D-dimensional latent space, from which latents are drawn as $\boldsymbol{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_D)$. A velocity predictor v_θ transports a class latent $\boldsymbol{z}_0$ toward a sample latent $\boldsymbol{z}_1$ along the linear interpolant $\boldsymbol{z}_t = (1-t)\,\boldsymbol{z}_0 + t\,\boldsymbol{z}_1$, where t is the flow time (distinct from the frame count T), trained with the conditional flow-matching loss

$$\mathcal{L}_{\mathrm{fm}} = \mathbb{E}_{t,\, c\in\mathcal{S},\, \boldsymbol{x}\in\mathcal{X}_c} \left\| v_\theta\big(\boldsymbol{z}_t, t\big) - \big(\boldsymbol{z}_1 - \boldsymbol{z}_0\big) \right\|^2, \tag{1}$$

where $\boldsymbol{z}_0$ is drawn from the conditional of class c, and $\boldsymbol{z}_1$ is drawn from the skeleton-side conditional whose parameters are predicted by $E^s(\boldsymbol{e})$, for a training sample of that class. The full objective $\mathcal{L}_{\mathrm{base}}$ additionally contains the reconstruction and KL regularization of the two encoders and a contrastive term on the velocity field, and a decoder D^s reconstructs skeleton embeddings from latents. At test time, the conditional of every candidate class is predicted from its semantics alone, the class whose latent best explains the test encoding under the velocity field is selected, and in GZSL a ratio gate on the seen and unseen errors first decides the domain of the sample (Eq. (12)).

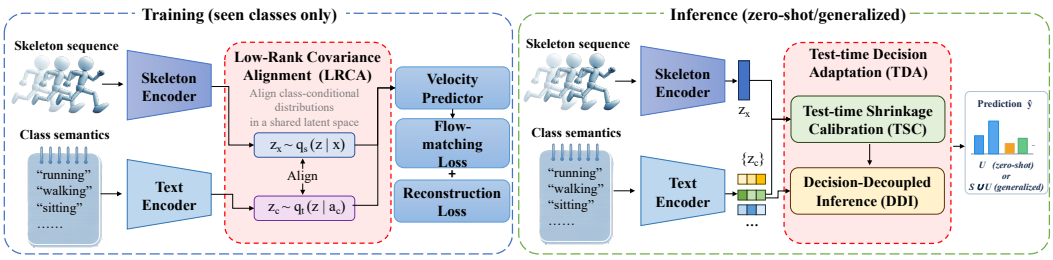


Figure 2: **Overview of LoAD.** Training (left): LRCA aligns the two modalities at the level of diagonal-plus-low-rank class conditionals. Inference (right): with all modules frozen, TDA calibrates the class latents with the unlabeled test batch (TSC) and decides with two complementary evidences (DDI).

3.2 Overview

Figure 2 gives an overview of LoAD. During training, v_θ is optimized with Eq. (1), and Low-Rank Covariance Alignment (LRCA, Sec. 3.3) extends each class conditional to a diagonal-plus-low-rank form and aligns the low-rank factors of the two modalities. At inference, all modules stay frozen, and Test-time Decision Adaptation (TDA, Sec. 3.4) adapts the decision rather than the parameters, where Test-time Shrinkage Calibration (TSC) re-anchors the class latents with statistics of the unlabeled test batch and Decision-Decoupled Inference (DDI) scores each class with two complementary evidences.

3.3 Low-Rank Covariance Alignment

The baseline models each class conditional as a diagonal Gaussian and aligns only means and per-dimension variances on seen classes. A diagonal covariance sets inter-dimension correlations to zero, so two conditionals may agree in means and variances yet differ in correlation structure. On seen classes, skeleton data can at least expose such a mismatch; on unseen classes, whose conditionals come from semantics alone, it can be neither detected nor corrected.

LRCA therefore extends both encoders with a low-rank shape component. Given a semantic embedding $\boldsymbol{a}_c$ or a skeleton embedding $\boldsymbol{e} = f(\boldsymbol{x})$, each encoder produces a hidden state $\boldsymbol{h} \in \mathbb{R}^D$ from which three linear heads emit the mean $\boldsymbol{\mu}$, the log-variances $\log \boldsymbol{\sigma}^2$, and a low-rank factor $\boldsymbol{L} \in \mathbb{R}^{D \times r}$ with $r \ll D$. A latent is drawn as

$$\boldsymbol{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}_1 + \boldsymbol{L}\,\boldsymbol{\epsilon}_2, \quad \boldsymbol{\epsilon}_1 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_D), \;\; \boldsymbol{\epsilon}_2 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_r), \tag{2}$$

where $\boldsymbol{\sigma} = \exp(\frac{1}{2}\log \boldsymbol{\sigma}^2)$ and $\odot$ denotes element-wise multiplication, which is equivalent to sampling from

$$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \qquad \boldsymbol{\Sigma} = \operatorname{diag}(\boldsymbol{\sigma}^2) + \boldsymbol{L}\boldsymbol{L}^\top. \tag{3}$$

Each column of $\boldsymbol{L}$ is a direction along which many latent dimensions move together, information that a diagonal covariance cannot carry. The shape factors of the two modalities are aligned on seen classes by

$$\mathcal{L}_{\text{shape}} = \mathbb{E}_{c \in \mathcal{S},\, \boldsymbol{x} \in \mathcal{X}_c} \left\| \boldsymbol{L}^t(\boldsymbol{a}_c) - \boldsymbol{L}^s(f(\boldsymbol{x})) \right\|_F^2, \tag{4}$$

where $\|\cdot\|_F$ is the Frobenius norm, so that the shape of an unseen class can be predicted from its semantics at test time. The full objective adds this single term,

$$\mathcal{L} = \mathcal{L}_{\text{base}} + \lambda_{\text{shape}}\, \mathcal{L}_{\text{shape}}, \tag{5}$$

where $\mathcal{L}_{\text{base}}$ already contains the baseline mean and variance alignment.

Two choices keep the component conservative. The new head is initialized near zero, so $\boldsymbol{L}\boldsymbol{L}^\top$ vanishes at the start of training and the model coincides exactly with the diagonal baseline, introducing shape structure only where seen data supports it. The low rank also keeps the overhead negligible, r vectors of length D per sample instead of the $D(D{+}1)/2$ entries of a full covariance. At inference, the conditional of every candidate class, seen or unseen, is predicted by feeding its semantics to E^t, and both the class latents and the test encoding are drawn with the full covariance of Eq. (3); how these latents are scored is deferred to Sec. 3.4. Figure 3 illustrates the design.

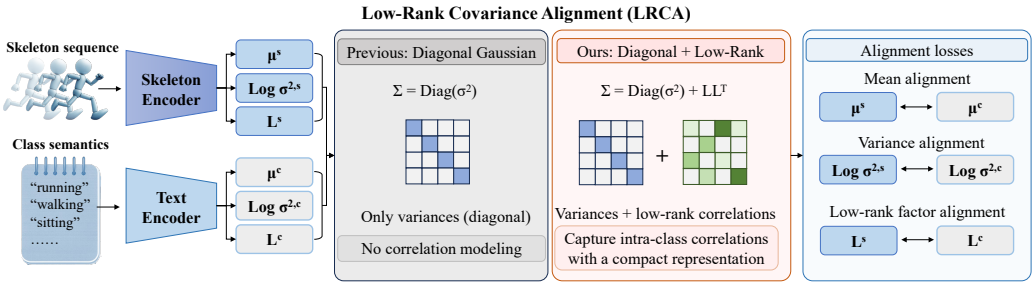


Figure 3: **Low-Rank Covariance Alignment (LRCA).** Each encoder outputs a mean, diagonal log-variances, and a low-rank factor $\boldsymbol{L}$, replacing the diagonal class covariance of the baseline (gray) with $\boldsymbol{\Sigma} = \mathrm{diag}(\boldsymbol{\sigma}^2) + \boldsymbol{L}\boldsymbol{L}^\top$ (red) to capture intra-class correlations compactly with $r \ll D$.

3.4 Test-time Decision Adaptation

Two sources of error remain after LRCA. First, the latent of an unseen class is predicted from semantics alone and may sit away from where its test samples fall. Second, the baseline scores each class with a single velocity error estimated at one flow time from one shared latent draw, so the ranking inherits the sampling noise of that draw. We address both by adapting the decision rather than the parameters of the frozen model with TDA, where TSC (Figure 4(a)) re-anchors the class latents and DDI (Figure 4(b)) then decides by fusing two complementary evidences. Algorithm 1 in the appendix summarizes the complete procedure, and the two components are described below.

Test-time shrinkage calibration. Let $\boldsymbol{x}_i$ be a test sample with embedding $\boldsymbol{e}_i = f(\boldsymbol{x}_i)$, and let $\boldsymbol{z}_{1i}$ be drawn via Eq. (2) from the conditional predicted by $E^s(\boldsymbol{e}_i)$. TDA scores a class against a test sample with the velocity error of the baseline, estimated more stably by averaging over a grid $\mathcal{T}$ of flow times and drawing an independent latent for every class,

$$\mathcal{E}_{ic}^{\mathrm{fm}} = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \left\| v_\theta(\boldsymbol{z}_t, t) - (\boldsymbol{z}_{1i} - \boldsymbol{z}_0^c) \right\|^2, \tag{6}$$

where $\boldsymbol{z}_t = (1-t)\,\boldsymbol{z}_0^c + t\,\boldsymbol{z}_{1i}$ is the interpolant of Sec. 3.1 and $\boldsymbol{z}_0^c \sim \mathcal{N}(\boldsymbol{\mu}_c^t, \boldsymbol{\Sigma}_c^t)$ is drawn from the conditional that E^t predicts from the semantics $\boldsymbol{a}_c$, of the low-rank form in Eq. (3), so the noise is not shared across classes. TSC converts the first-pass errors of Eq. (6) into soft votes and pools them into per-class anchors,

$$w_{ic} = \frac{\exp(-\mathcal{E}_{ic}^{\mathrm{fm}}/\tau)}{\sum_{c'} \exp(-\mathcal{E}_{ic'}^{\mathrm{fm}}/\tau)}, \quad \hat{\boldsymbol{\mu}}_c = \frac{\sum_i w_{ic}\,\boldsymbol{\mu}_i^s}{\sum_i w_{ic}}, \quad \hat{\boldsymbol{\sigma}}_c^2 = \frac{\sum_i w_{ic}\,(\boldsymbol{\mu}_i^s - \hat{\boldsymbol{\mu}}_c)^2}{\sum_i w_{ic}}, \tag{7}$$

where $\boldsymbol{\mu}_i^s$ is the posterior mean of sample i predicted by E^s and all operations are element-wise. Each row of w_{ic} sums to one, so $w_c = \sum_i w_{ic}$ counts the votes of class c. Anchors supported by few votes can be noisy and are therefore shrunk toward the statistics of the batch in proportion to their vote count,

$$\alpha_c = \frac{1}{1 + w_c}, \quad \hat{\boldsymbol{\mu}}_c \leftarrow \alpha_c\,\bar{\boldsymbol{\mu}}^s + (1-\alpha_c)\,\hat{\boldsymbol{\mu}}_c, \quad \hat{\boldsymbol{\sigma}}_c^2 \leftarrow \alpha_c\,\bar{\boldsymbol{\sigma}}^{2\,s} + (1-\alpha_c)\,\hat{\boldsymbol{\sigma}}_c^2, \tag{8}$$

where $\bar{\boldsymbol{\mu}}^s$ and $\bar{\boldsymbol{\sigma}}^{2\,s}$ are the mean and element-wise variance of $\{\boldsymbol{\mu}_i^s\}$ over the batch. A class with many votes keeps its own anchor, and one with few falls back to the batch statistics. Every class latent of $\mathcal{S} \cup \mathcal{U}$ is then re-anchored,

$$\tilde{\boldsymbol{z}}_0^c = (\boldsymbol{z}_0^c - \bar{\boldsymbol{\mu}}^t) \odot \left(\hat{\boldsymbol{\sigma}}_c^2 / \bar{\boldsymbol{\sigma}}^{2\,t}\right)^{1/2} + \hat{\boldsymbol{\mu}}_c, \tag{9}$$

where $\bar{\boldsymbol{\mu}}^t$ and $\bar{\boldsymbol{\sigma}}^{2\,t}$ are the cross-class statistics of the text-predicted means. The offset from $\bar{\boldsymbol{\mu}}^t$ preserves the relative layout of the classes, which is what semantics provides, while the anchor sets the absolute position with batch evidence. The velocity errors are recomputed once with $\tilde{\boldsymbol{z}}_0^c$ held fixed, without new draws. TSC is closed-form, label-free, and adds a single deterministic scoring pass. We apply it when $|\mathcal{S}| \le K$, since with many candidates each vote spreads thinly, α_c approaches one, and calibration reduces to the batch statistics.

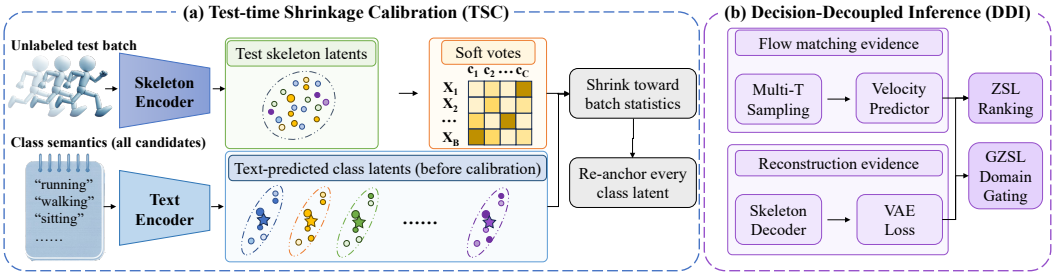


Figure 4: **Test-time Decision Adaptation (TDA).** (a) Test-time shrinkage calibration encodes the unlabeled test batch, pools its samples into per-class anchors by soft votes, and shrinks every text-predicted class latent toward the batch statistics. (b) Decision-decoupled inference scores each class with two evidences, a multi-time flow-matching velocity error and a deterministic reconstruction error, combined by ranking for ZSL and by domain gating for GZSL.

Decision-decoupled inference. DDI decides with the calibrated velocity error of Eq. (6) as the first evidence. The second evidence decodes the class mean and compares it with the test embedding,

$$\mathcal{E}_{ic}^{\mathrm{vae}} = \left\| D^s(\boldsymbol{\mu}_c^t) - \boldsymbol{e}_i \right\|^2, \tag{10}$$

which involves no sampling. In ZSL the two evidences are fused,

$$\hat{y}_i = \arg\min_{c \in \mathcal{U}} \; \mathcal{E}_{ic}^{\mathrm{fm}} + \lambda \, \mathcal{E}_{ic}^{\mathrm{vae}}, \tag{11}$$

where $\lambda = 0$ recovers the baseline decision. In GZSL the ratio gate of the baseline is kept on the velocity errors alone, since its threshold was calibrated for that quantity,

$$r_i = \frac{\min_{c \in \mathcal{S}} \mathcal{E}_{ic}^{\mathrm{fm}}}{\min_{c \in \mathcal{U}} \mathcal{E}_{ic}^{\mathrm{fm}}}, \qquad \hat{y}_i = \begin{cases} \arg\min_{c \in \mathcal{U}} \mathcal{E}_{ic}^{\mathrm{fm}}, & r_i > \gamma, \\ \arg\min_{c \in \mathcal{S}} \mathcal{E}_{ic}^{\mathrm{fm}}, & r_i \le \gamma. \end{cases} \tag{12}$$

The reconstruction evidence thus enters only the ZSL ranking. Unless stated otherwise, $\tau = 1$ and $K = 60$; the fusion weight λ of each split is listed in Table 7 in the appendix.

4 Experiments

4.1 Experimental Setup

We evaluate on NTU RGB+D 60 Shahroudy et al. (2016) (60 classes, cross-subject 55/5 and 48/12 seen/unseen splits), NTU RGB+D 120 Liu et al. (2020) (120 classes, 110/10 and 96/24 splits under the same protocol), and PKU-MMD Chunhui et al. (2017) (51 classes, 46/5 split). We follow the evaluation protocol of Flora Chen et al. (2026b) and reproduce it with its official implementation. Skeleton sequences are represented by the released SynSE-benchmark Shift-GCN features ($d_e = 256$), and class semantics come from CLIP ViT-L/14@336. TSC uses statistics over the whole unlabeled test set, and all remaining hyperparameters are listed in Table 7 in the appendix.

4.2 Comparison with State-of-the-Art Methods

NTU RGB+D 60/120. Table 1 summarizes the results across all four seen/unseen splits under both ZSL and GZSL protocols. LoAD improves over its direct baseline Flora* on the ZSL accuracy of every split, with gains of $+1.8$, $+2.4$, $+1.1$, and $+1.9$ on 55/5, 48/12, 110/10, and 96/24, respectively. Under the GZSL protocol, LoAD improves the unseen-class accuracy by up to $+3.0$ on the two NTU RGB+D 120 splits, while the harmonic mean stays comparable or better on all four splits (within 0.1 on 48/12 and improved elsewhere), indicating that the gains do not come from a seen/unseen trade-off. Methods that rely on larger training budgets or self-pretrained backbones remain stronger on several GZSL columns. LoAD is orthogonal to these directions, as it introduces neither extra data nor backbone pretraining, and its test-time adaptation is closed-form and gradient-free.

PKU-MMD. As shown in Table 2, LoAD surpasses the reproduced Flora* by $+3.3$ ZSL accuracy and $+1.1$ average on the 46/5 split, with results averaged over three random splits.

Table 1: Comparison with state-of-the-art methods on NTU RGB+D 60 and 120 (Xsub) under ZSL and GZSL (S/U/H) protocols. The best, second-best, and third-best results in each column are marked in **red**, **blue**, and **green**, respectively. † denotes methods using the released SynSE-based Shift-GCN features Gupta et al. (2021). ‡ denotes two-stream fusion. * denotes our reproduction of the direct baseline using the official implementation under the same protocol; all other results are taken from the original papers.

Method	Publication	NTU RGB+D 60								NTU RGB+D 120							
		55/5				48/12				110/10				96/24			
		ZSL	S	U	H	ZSL	S	U	H	ZSL	S	U	H	ZSL	S	U	H
ReViSE Hubert Tsai et al. (2017)	ICCV'17	69.5	40.8	50.2	45.0	24.0	21.8	14.8	17.6	19.8	0.6	14.5	1.1	8.5	3.4	1.5	2.1
JPoSE Wray et al. (2019)	ICCV'19	73.7	66.5	53.5	59.3	27.5	28.6	18.7	22.6	57.3	53.6	11.6	19.1	38.1	41.0	3.8	6.9
CADA-VAE Schönfeld et al. (2019)	CVPR'19	76.9	56.1	56.0	56.0	32.1	50.4	25.0	33.4	52.5	50.2	43.9	46.8	38.7	48.3	27.5	35.1
SynSE Gupta et al. (2021)	ICIP'21	71.9	51.3	47.4	49.2	31.3	44.1	22.9	30.1	52.4	57.3	43.2	49.5	41.9	48.1	32.9	39.1
GZSSAR† Li et al. (2023)	ICIG'23	83.6	71.7	66.2	68.8	49.2	58.8	40.0	47.6	71.2	46.8	68.3	55.6	59.7	56.8	48.6	52.4
SMIE Zhou et al. (2023)	ACMMM'23	77.9	-	-	-	41.5	-	-	-	61.3	-	-	-	42.3	-	-	-
PURLS Zhu et al. (2024)	CVPR'24	79.2	-	-	-	41.0	-	-	-	72.0	-	-	-	52.0	-	-	-
SA-DVAE Li et al. (2024)	ECCV'24	82.4	62.8	70.8	66.3	41.4	50.2	36.9	42.6	68.8	61.1	59.8	60.4	46.1	58.8	35.8	44.5
STAR Chen et al. (2024)	ACMMM'24	81.4	69.0	69.9	69.4	45.1	62.7	37.0	46.6	63.3	59.9	52.7	56.1	44.3	51.2	36.9	42.9
DVTA† Kuang et al. (2026)	PR'25	79.3	-	-	-	44.1	-	-	-	**74.9**	-	-	-	51.8	-	-	-
InfoCPL† Xu et al. (2025)	TMM'25	85.9	-	-	-	53.3	-	-	-	74.8	-	-	-	60.1	-	-	-
SCoPLe† Zhu et al. (2025a)	CVPR'25	84.1	69.6	71.9	70.8	53.0	54.5	**61.8**	57.9	74.5	63.5	61.1	62.3	52.2	53.3	51.2	52.2
Neuron‡ Chen et al. (2025)	CVPR'25	86.9	69.1	73.8	71.4	62.7	61.6	**56.8**	59.1	71.5	**67.6**	59.5	63.3	57.1	**67.5**	44.4	**53.6**
FS-VAE† Wu et al. (2025)	ICCV'25	86.9	**77.0**	74.5	75.7	57.2	56.2	48.6	52.1	74.4	59.2	**67.9**	63.3	62.5	57.8	**51.9**	**54.7**
TDSM† Do & Kim (2025)	ICCV'25	86.5	-	-	-	56.0	-	-	-	74.2	-	-	-	**65.1**	-	-	-
SkeletonCache Zhu et al. (2025b)	NeurIPS'25	**89.4**	64.6	**79.2**	71.2	47.8	53.2	44.4	48.4	74.3	**62.2**	65.8	**63.9**	53.1	**59.9**	44.1	50.8
STAR++ Chen et al. (2026a)	TCSVT'26	84.4	61.1	73.6	66.8	49.5	58.2	40.4	47.7	72.0	59.0	55.4	57.2	53.5	52.8	45.2	48.7
SCALEOraki et al. (2027)	ICPR'26	84.5	-	-	-	50.0	-	-	-	73.6	-	-	-	54.9	-	-	-
SkeletonContext Wang et al. (2026)	CVPR'26	**89.6**	**78.3**	76.1	**77.1**	**64.4**	**65.5**	**57.1**	**61.1**	74.2	61.1	65.5	63.1	60.1	**64.6**	49.6	**56.1**
Flora†* Chen et al. (2026b)	CVPR'26	85.4	75.8	**77.1**	**76.5**	**63.7**	**67.7**	53.5	**59.8**	**79.3**	**63.7**	**68.9**	**66.2**	**64.9**	53.7	**51.6**	52.6
LoAD† (Ours)	–	**87.2**	**77.6**	**76.3**	**77.1**	**66.1**	**66.3**	54.3	**59.7**	**80.4**	62.1	**71.8**	**66.6**	**66.8**	51.3	**54.6**	52.9

Table 2: Comparison on PKU-MMD (46/5) under ZSL and GZSL (H) protocols, averaged over three random splits following Li et al. (2024). Bold and underline denote the best and second-best results. All listed methods are evaluated on publicly released features.

Method	Publication	ZSL	H
ReViSE Hubert Tsai et al. (2017)	ICCV'17	59.3	49.8
JPoSE Wray et al. (2019)	ICCV'19	57.2	51.6
CADA-VAE Schönfeld et al. (2019)	CVPR'19	60.7	45.8
SynSE Gupta et al. (2021)	ICIP'21	53.9	49.5
SMIE Zhou et al. (2023)	MM'23	60.8	–
SA-DVAE Li et al. (2024)	ECCV'24	66.5	54.7
SCoPLe Zhu et al. (2025a)	CVPR'25	<u>71.4</u>	54.9
TDSM Do & Kim (2025)	ICCV'25	70.8	–
FS-VAE Wu et al. (2025)	ICCV'25	71.2	**59.0**
SkeletonCache Zhu et al. (2025b)	NeurIPS'25	71.1	58.5
Flora* Chen et al. (2026b)	CVPR'26	69.0	57.8
LoAD (Ours)	–	**72.3**	<u>58.9</u>

4.3 Ablation Studies

All ablation runs are our own, conducted under the identical pipeline, features, and training budget; the default configuration is shaded gray where applicable, and we report ZSL accuracy unless stated otherwise.

Contribution of each component. We progressively enable LRCA, TSC, and DDI on the Flora baseline, with results reported in Table 3. The full model improves over the baseline on all four splits, by $+1.8$, $+1.4$, $+1.1$, and $+1.9$ on 55/5, 48/12, 110/10, and 96/24, respectively. LRCA alone improves 55/5, 110/10, and 96/24 by $+1.0$, $+1.9$, and $+0.8$, while on 48/12 the gain comes from the test-time modules alone, with +TSC and +DDI bringing a further $+1.0$ and $+0.4$. Since TSC is gated by the seen-class count (Sec. 3.4), the +TSC row coincides with the +LRCA row on 110/10 and 96/24 by design. On 110/10, DDI lowers the ZSL accuracy by 0.8 but improves the GZSL unseen accuracy over the baseline by $+2.9$ (Table 1), and we retain it for this GZSL gain.

Table 3: Progressive ablation on NTU RGB+D 60/120 (Xsub, ZSL accuracy). TSC is gated by the seen-class count (Sec. 3.4), so the +LRCA and +TSC rows coincide on 110/10 and 96/24 by design.

	NTU RGB+D 60		NTU RGB+D 120	
Configuration	55/5	48/12	110/10	96/24
Baseline (Flora pipeline)	85.4	63.7	79.3	64.9
+ LRCA	86.4	64.7	81.2	65.7
+ TSC	86.8	65.7	81.2	65.7
+ DDI (full LoAD)	87.2	66.1	80.4	66.8

Table 4: Sensitivity of LoAD to the rank r and the alignment weight λ_{shape} on the four NTU splits (ZSL accuracy); defaults in gray. Within each split the grid varies by at most 2.8 pp.

	NTU-60		NTU-120	
	55/5 (u5)	48/12 (u12)	110/10 (u10)	96/24 (u24)
r=1	87.46	65.99	79.63	65.23
r=2	86.65	63.58	80.19	67.17
r=4	87.17	66.06	80.38	66.81
r=8	86.87	65.54	79.77	64.73
r=16	86.80	63.30	79.79	66.73
λ=0.02	86.80	65.47	79.30	66.18
λ=0.05	86.73	65.54	79.44	66.07
λ=0.1	87.17	66.06	80.38	66.81
λ=0.2	86.73	65.35	79.42	66.17
λ=0.5	87.02	65.35	79.46	66.26

TSC across seen-set sizes. To isolate the effect of TSC, we toggle it on top of the full model while keeping every other module fixed, and report the results in Table 5. TSC re-anchors class latents with soft votes from the test batch, and with many candidate classes the votes spread thin and the anchors fall back to the batch statistics (Sec. 3.4), so TSC should help on small seen sets and do little on large ones. The results follow this pattern: TSC improves ZSL accuracy on every small seen set ($+1.26$ on 55/5, $+3.61$ on 48/12, $+0.92$ on PKU-MMD 46/5) and does not help on the large ones (-0.82 on 110/10, -0.20 on 96/24), with the same sign pattern in GZSL-H on the NTU splits; the only exception is a -0.09 change in GZSL-H on PKU-MMD, which we accept rather than introduce per-split choices. LoAD therefore treats TSC as a conditional component, switched by the single rule $|\mathcal{S}| \leq K$ with $K = 60$ rather than by per-split choices, and the negative cells in Table 5 are what motivate this gate.

Coupling between the test-time modules. The two test-time modules are coupled. Toggling TSC on top of the full pipeline shifts 48/12 by $+3.61$ and 55/5 by $+1.26$ (Table 5), whereas in the chain without DDI the same toggle brings only $+1.0$ and $+0.4$ (Table 3); DDI thus amplifies the effect of TSC. Conversely, disabling TSC in the full model drops 48/12 to 62.45, below the 64.7 baseline, so on small seen sets the two modules need to act together, which the gate enforces.

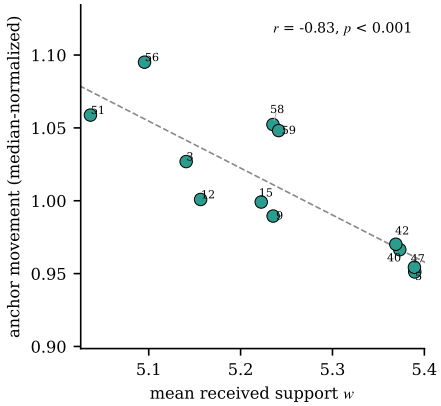
Hyper-parameter sensitivity. As shown in Table 4, we sweep the rank $r \in \{1, 2, 4, 8, 16\}$ and the alignment weight $\lambda_{\text{shape}} \in \{0.02, 0.05, 0.1, 0.2, 0.5\}$ across all four splits. Within each split, every grid varies by at most 2.8 pp, and the defaults (r=4, λ_{shape}=0.1) lie within 0.4 pp of the best cell on every split. The same defaults serve all datasets and splits.

4.4 Visualization

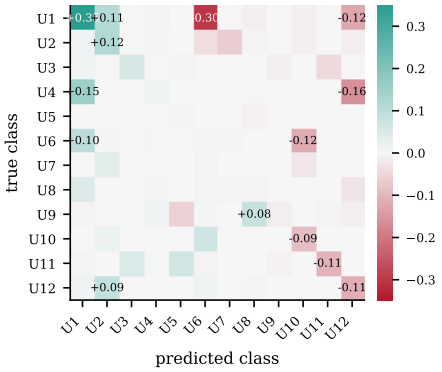
Adaptivity of TSC. In Figure 5a, we relate the anchor movement of each unseen class to the support w it receives from the test batch on the 48/12 split. The two are strongly anti-correlated (Pearson $r = -0.83$, $p < 0.001$, $n = 12$): weakly supported anchors are updated substantially, while well-supported ones are left nearly intact, as expected from the shrinkage schedule $\alpha = 1/(1+w)$ (Sec. 3.4). The correction is thus largest precisely where the current anchor is least supported by the test batch.

Table 5: Effect of TSC on the full model, with all other modules fixed (Δ = ON $-$ OFF). The ON column applies TSC regardless of the gate; on 110/10 and 96/24 the gate disables TSC (Sec. 3.4), so the OFF column coincides with the final model. The PKU-MMD row reports one of the three random splits.

			ZSL			GZSL-H		
Dataset	Split	Seen	TSC OFF	TSC ON	Δ	TSC OFF	TSC ON	Δ
NTU RGB+D 60	55/5	55	85.91	87.17	+1.26	76.34	77.09	+0.75
NTU RGB+D 60	48/12	48	62.45	66.06	+3.61	58.36	59.68	+1.32
NTU RGB+D 120	110/10	110	80.38	79.56	-0.82	66.60	65.09	-1.51
NTU RGB+D 120	96/24	96	66.81	66.61	-0.20	52.87	51.58	-1.29
PKU-MMD	46/5	46	68.83	69.75	+0.92	57.64	57.55	-0.09



(a) TSC calibrates each class anchor in proportion to its need. Each point is one unseen class; classes receiving less support w from the test batch undergo larger anchor movements (Pearson $r = -0.83$, $p < 0.001$, movement median-normalized). The dashed line is a least-squares fit.



(b) Confusion difference between LoAD and the reproduced Flora pipeline, row-normalized per true class. Diagonal entries are per-class accuracy changes and off-diagonal entries are changes in confusion mass. Classes U1 to U12 are sorted by decreasing accuracy gain.

Figure 5: Mechanistic and error analyses of LoAD on the 48/12 split of NTU RGB+D 60.

Per-class error analysis. Figure 5b compares the per-class confusion of LoAD and the Flora* baseline on the same split. Nine of the twelve unseen classes improve, and the largest gain (U1, "move the hip backward and down to knee horizon", $+0.35$) coincides with a resolved confusion with the semantically adjacent U6 ("bend knees and hips to move the hand down to the ground then pinch then get up", -0.30): both classes involve lowering the hip while moving the hand. This suggests that LoAD mainly repairs confusions between semantically close actions. These gains are accompanied by two new confusion pairs, each changing by at most $+0.15$, and no unseen class loses more than 0.11 accuracy.

5 Conclusion

We present LoAD, which strengthens both phases of the align-then-classify pipeline for zero-shot skeleton action recognition. LRCA upgrades the class conditionals from diagonal Gaussians to a diagonal-plus-low-rank form and aligns their correlation structures, while TDA adapts the decision function of the frozen model. DDI fuses two complementary sources of evidence, and TSC re-anchors class latents using the unlabeled test batch. Both TDA components are closed-form and require no gradient-based optimization at test time. Extensive experiments on NTU RGB+D 60/120 and PKU-MMD under ZSL and GZSL show that LoAD achieves competitive performance, highlighting the effectiveness and simplicity of the proposed framework.

AI USE STATEMENT

In this work, we used generative AI tools for language editing only, namely polishing the English of the paper to improve readability. We did not use generative AI tools for any task in the required-disclosure list, such as formulating hypotheses, designing the methodology, implementing the method, or interpreting results. The remaining items in the required-disclosure list are not applicable to this work.

We have reviewed all AI-assisted edits and take responsibility for the final content of the paper, including the text produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All experiments use publicly released SynSE-benchmark Shift-GCN features and standard splits of NTU RGB+D 60/120 and PKU-MMD, and all hyper-parameters are summarized in Table 7 in the appendix.

REFERENCES

Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *CVPR*, June 2015.

Wei-Lun Chao, Soravit Changpinyo, Boqing Gong, and Fei Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (eds.), *ECCV*, pp. 52–68, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46475-6.

Yang Chen, Jingcai Guo, Tian He, Xiaocheng Lu, and Ling Wang. Fine-grained side information guided dual-prompts for zero-shot skeleton action recognition. In *ACM MM*, pp. 778–786, 2024.

Yang Chen, Jingcai Guo, Song Guo, and Dacheng Tao. Neuron: Learning context-aware evolving representations for zero-shot skeleton action recognition. In *CVPR*, pp. 8721–8730, June 2025.

Yang Chen, Jingcai Guo, Miaoge Li, Zhijie Rao, and Song Guo. Star++: Region-aware conditional semantics via interpretable side information for zero-shot skeleton action recognition. *TCSVT*, 36 (5):7168–7183, 2026a. doi: 10.1109/TCSVT.2026.3651695.

Yang Chen, Miaoge Li, Zhijie Rao, Deze Zeng, Song Guo, and Jingcai Guo. Learning by neighbor-aware semantics, deciding by open-form flows: Towards robust zero-shot skeleton action recognition. In *CVPR Findings*, pp. 3374–3383, June 2026b.

Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *ICCV*, pp. 13359–13368, 2021.

Ke Cheng, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with shift graph convolutional network. In *CVPR*, pp. 180–189, 2020. doi: 10.1109/CVPR42600.2020.00026.

Liu Chunhui, Hu Yueyu, Li Yanghao, Song Sijie, and Liu Jiaying. Pku-mmd: A large scale benchmark for continuous multi-modal human action understanding. *ACM Multimedia workshop*, 2017.

Jeonghyeok Do and Munchurl Kim. Bridging the skeleton-text modality gap: Diffusion-powered modality alignment for zero-shot skeleton-based action recognition. In *ICCV*, pp. 12757–12768, October 2025.

Andrea Frome, Greg S. Corrado, Jon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *NeurIPS*, 2013.

Yanwei Fu, Timothy M Hospedales, Tao Xiang, and Shaogang Gong. Transductive multi-view zero-shot learning. *TPAMI*, 37(11):2332–2345, 2015.

Pranay Gupta, Divyanshu Sharma, and Ravi Kiran Sarvadevabhatla. Syntactically guided generative embeddings for zero-shot skeleton action recognition. In *ICIP*, pp. 439–443, 2021. doi: 10.1109/ICIP42928.2021.9506179.

Yao-Hung Hubert Tsai, Liang-Kang Huang, and Ruslan Salakhutdinov. Learning robust visual-semantic embeddings. In *ICCV*, Oct 2017.

Yusuke Iwasawa and Yutaka Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. In *NeurIPS*, 2021.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

Jidong Kuang, Hongsong Wang, Chaolei Han, Yang Zhang, and Jie Gui. Zero-shot skeleton-based action recognition with dual visual-text alignment. *PR*, 171:112342, 2026. ISSN 0031-3203.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of multivariate analysis*, 88(2):365–411, 2004.

Ming-Zhe Li, Zhen Jia, Zhang Zhang, Zhanyu Ma, and Liang Wang. Multi-semantic fusion model for generalized zero-shot skeleton-based action recognition. In *ICIG*, pp. 68–80. Springer, 2023.

Sheng-Wei Li, Zi-Xiang Wei, Wei-Jie Chen, Yi-Hsin Yu, Chih-Yuan Yang, and Jane Yung jen Hsu. Sa-dvae: Improving zero-shot skeleton-based action recognition by disentangled variational autoencoders. In *ECCV*, 2024.

Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *ICML*, pp. 6028–6039. PMLR, 2020.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *ICLR*, 2023.

Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C. Kot. Ntu rgb+d 120: A large-scale benchmark for 3d human activity understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(10):2684–2701, 2020. doi: 10.1109/TPAMI.2019.2916873.

Soroush Oraki, Feng Ding, and Jie Liang. Scale: Semantic- and confidence-aware conditional variational autoencoder for zero-shot skeleton-based action recognition. In Maria De Marsico, Tin Kam Ho, Frederic Jurie, Cheng-Lin Liu, Daniel Lopresti, Ingela Nyström, Jean-Marc Ogier, Arun Ross, and Liang Wang (eds.), *Pattern Recognition*, pp. 248–263, Cham, 2027. Springer Nature Switzerland. ISBN 978-3-032-31920-3.

Farhad Pourpanah, Moloud Abdar, Yuxuan Luo, Xinlei Zhou, Ran Wang, Chee Peng Lim, Xi-Zhao Wang, and Q. M. Jonathan Wu. A review of generalized zero-shot learning methods. *TPAMI*, 45 (4):4051–4070, 2023. doi: 10.1109/TPAMI.2022.3191696.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021.

Edgar Schönfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero- and few-shot learning via aligned variational autoencoders. In *CVPR*, pp. 8239–8247, 2019. doi: 10.1109/CVPR.2019.00844.

Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *CVPR*, pp. 1010–1019, 2016. doi: 10.1109/CVPR.2016.115.

Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *CVPR*, pp. 12026–12035, 2019.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *ICML*, 2020.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*, 2021.

Ning Wang, Tieyue Wu, Naeha Sharif, Farid Boussaid, Guangming Zhu, Lin Mei, Mohammed Bennamoun, and Liang Zhang. Skeletoncontext: Skeleton-side context prompt learning for zero-shot skeleton-based action recognition. In *CVPR*, pp. 20170–20180, 2026.

Michael Wray, Diane Larlus, Gabriela Csurka, and Dima Damen. Fine-grained action retrieval through multiple parts-of-speech embeddings. In *ICCV*, October 2019.

Wenhan Wu, Zhishuai Guo, Chen Chen, Hongfei Xue, and Aidong Lu. Frequency-semantic enhanced variational autoencoder for zero-shot skeleton-based action recognition. In *ICCV*, pp. 11122–11131, October 2025.

Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. volume 41, pp. 2251–2265, 2019a. doi: 10.1109/TPAMI.2018.2857768.

Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. volume 41, pp. 2251–2265, 2019b. doi: 10.1109/TPAMI.2018.2857768.

Yongqin Xian, Saurabh Sharma, Bernt Schiele, and Zeynep Akata. F-vaegan-d2: A feature generating framework for any-shot learning. In *CVPR*, June 2019c.

Haojun Xu, Yan Gao, Jie Li, and Xinbo Gao. An information compensation framework for zero-shot skeleton-based action recognition. *TMM*, 27:4882–4894, 2025.

Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*, 2018.

Yujie Zhou, Wenwen Qiang, Anyi Rao, Ning Lin, Bing Su, and Jiaqi Wang. Zero-shot skeleton-based action recognition via mutual information estimation and maximization. In *ACM MM*, pp. 5302–5310, 2023.

Anqi Zhu, Qiuhong Ke, Mingming Gong, and James Bailey. Part-aware unified representation of language and skeleton for zero-shot action recognition. In *CVPR*, pp. 18761–18770, 2024. doi: 10.1109/CVPR52733.2024.01775.

Anqi Zhu, Jingmin Zhu, James Bailey, Mingming Gong, and Qiuhong Ke. Semantic-guided cross-modal prompt learning for skeleton-based zero-shot action recognition. In *CVPR*, pp. 13876–13885, 2025a. doi: 10.1109/CVPR52734.2025.01295.

Jingmin Zhu, Anqi Zhu, Hossein Rahmani, Jun Liu, Mohammed Bennamoun, and Qiuhong Ke. Boosting skeleton-based zero-shot action recognition with training-free test-time adaptation. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *NeurIPS*, volume 38, Main Conference, pp. 103521–103551. Curran Associates, Inc., 2025b. doi: 10.52202/085713-3458.

Jingmin Zhu, Anqi Zhu, James Bailey, Jun Liu, Hossein Rahmani, Mohammed Bennamoun, Farid Boussaid, and Qiuhong Ke. Dynapurls: Dynamic refinement of part-aware representations for skeleton-based zero-shot action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(8):9659–9674, 2026. doi: 10.1109/TPAMI.2026.3680873.

A Appendix

A.1 Unseen Class Names of the 48/12 Split

Table 6 lists the twelve unseen classes of the NTU RGB+D 60 48/12 split, which are referred to as U1–U12 in the main paper. In the per-class analysis (Fig. 5b), the classes are additionally sorted by decreasing accuracy gain, so the position of a class in that figure does not correspond to its index here. Several of the names share near-identical body-part templates: U4, U5, and U9 all involve

Table 6: The twelve unseen classes of the NTU RGB+D 60 48/12 split, referred to as U1–U12 in the main paper.

ID	Class name
U1	move the hip backward and down to knee horizon
U2	move the hand up to the head then move the hand up and down
U3	strike hands together repeatedly in front of the spine
U4	pinch and move the hand up to the head
U5	pinch in front of the spine and tremble the hand
U6	bend knees and hips to move the hand down to the ground then pinch then get up
U7	move the hand up to the head then tremble the hand
U8	fist and move the hand up and forward
U9	pinch in front of the spine and hold
U10	fist and move the hand up to the head then look up
U11	move the hip forward and up to knee vertical
U12	release the hand in front of the middle of the spine

pinching in front of the spine, U2, U7, and U10 share the prefix "move the hand up to the head", and U1 and U6 both describe lowering the hip while moving the hand. These semantic families are the source of the semantically adjacent confusions analyzed in Sec. 4.4.

A.2 Test-time Decision Adaptation Algorithm

Algorithm 1 summarizes the complete procedure of TDA described in Sec. 3.4 of the main paper. The algorithm receives a test batch $\{\boldsymbol{x}_i\}_{i=1}^{B}$ and the semantics of all candidate classes, where the candidate set is $\mathcal{U}$ in ZSL and $\mathcal{S} \cup \mathcal{U}$ in GZSL. It first encodes the batch with the frozen skeleton encoder, draws class latents from the text-predicted conditionals, and computes first-pass velocity errors by multi-t averaging in Eq. (6). When the seen-class count satisfies $|\mathcal{S}| \leq K$, TSC converts these errors into soft votes, pools per-class anchors, shrinks them toward the batch statistics in proportion to their vote counts, re-anchors every class latent by Eq. (9), and recomputes the velocity errors once from the calibrated latents without new draws. The final decision follows the DDI rule of Sec. 3.4. In ZSL the ranking fuses the calibrated velocity error with the deterministic reconstruction error in Eq. (11), while in GZSL the ratio gate of Eq. (12) first decides the domain of the sample and then ranks within it. All steps are closed-form, run on the frozen model, and require no gradient updates.

A.3 Implementation and Hyper-parameter Details

Table 7 lists the training and inference hyper-parameters of LoAD. The defaults stated in Sec. 3.4 apply to all datasets, namely $\tau = 1$ and $K = 60$. The GZSL gate threshold γ is dataset-wise and the fusion weight λ is split-wise, and both are listed in the table. Training uses a single configuration for all benchmarks except the low-rank size r, which follows the dataset-wise setting in the table.

A.4 Computational Cost Analysis

Table 8 summarizes the cost of LoAD against the Flora pipeline, where the frozen CLIP text encoder and the skeleton backbone are shared across all methods.

Parameters. The only trainable additions are the low-rank heads of LRCA in the two modality encoders, which raise the trainable parameters from 15.81M to 18.96M ($+3.15$M, $+19.9\%$). TDA adds no parameters at all, since DDI reads its two evidences from the existing velocity predictor and decoder, and TSC pools its anchors in closed form from the frozen encoders, whose statistics account for a negligible share of the latency.

Inference cost. With $K = 12$ unseen classes, the baseline costs 8.25 GFLOPs per sample and LoAD costs 12.34 GFLOPs, a ratio of $1.50\times$. The increase comes from two components only, namely a single re-scoring pass of the velocity error after TSC calibration, which reuses the calibrated latents without resampling or re-encoding the test batch, and the deterministic reconstruction

Algorithm 1 Test-time Decision Adaptation (TDA)

Require: test batch $\{\boldsymbol{x}_i\}_{i=1}^{B}$, semantics $\{\boldsymbol{a}_c\}_{c\in\mathcal{C}}$ ($\mathcal{C} = \mathcal{U}$ for ZSL, $\mathcal{C} = \mathcal{S}\cup\mathcal{U}$ for GZSL), trained $f, E^t, E^s, D^s, v_\theta$

$\boldsymbol{e}_i \leftarrow f(\boldsymbol{x}_i)$; $(\boldsymbol{\mu}_i^s, \boldsymbol{\sigma}_i^s, \boldsymbol{L}_i^s) \leftarrow E^s(\boldsymbol{e}_i)$; draw $\boldsymbol{z}_{1i}$ via Eq. (2) ▷ encode test batch

$(\boldsymbol{\mu}_c^t, \boldsymbol{\sigma}_c^t, \boldsymbol{L}_c^t) \leftarrow E^t(\boldsymbol{a}_c)$ for all $c \in \mathcal{C}$ ▷ conditionals from text

Draw $\boldsymbol{z}_0^c \sim \mathcal{N}\big(\boldsymbol{\mu}_c^t, \mathrm{diag}\big((\boldsymbol{\sigma}_c^t)^2\big) + \boldsymbol{L}_c^t \boldsymbol{L}_c^{t\top}\big)$ for all c, and compute first-pass $\mathcal{E}_{ic}^{\mathrm{fm}}$ by multi-t averaging ▷ **DDI**, evidence 1

if $|\mathcal{S}| \le K$ **then** ▷ **TSC**, small seen set

  $w_{ic} \leftarrow \mathrm{softmax}_c\big(-\mathcal{E}_{ic}^{\mathrm{fm}}/\tau\big)$; $w_c \leftarrow \sum_i w_{ic}$ ▷ soft votes

  $\hat{\boldsymbol{\mu}}_c \leftarrow \sum_i w_{ic}\,\boldsymbol{\mu}_i^s / w_c$; $\hat{\boldsymbol{\sigma}}_c^2 \leftarrow \sum_i w_{ic}\,(\boldsymbol{\mu}_i^s - \hat{\boldsymbol{\mu}}_c)^2 / w_c$ ▷ anchors

  $\alpha_c \leftarrow 1/(1+w_c)$; $\hat{\boldsymbol{\mu}}_c \leftarrow \alpha_c\,\bar{\boldsymbol{\mu}}^s + (1-\alpha_c)\,\hat{\boldsymbol{\mu}}_c$; $\hat{\boldsymbol{\sigma}}_c^2 \leftarrow \alpha_c\,\bar{\boldsymbol{\sigma}}^{2\,s} + (1-\alpha_c)\,\hat{\boldsymbol{\sigma}}_c^2$ ▷ shrinkage

  $\boldsymbol{z}_0^c \leftarrow (\boldsymbol{z}_0^c - \bar{\boldsymbol{\mu}}^t) \odot \sqrt{\hat{\boldsymbol{\sigma}}_c^2 / \bar{\boldsymbol{\sigma}}^{2\,t}} + \hat{\boldsymbol{\mu}}_c$ ▷ re-anchor

  Recompute $\mathcal{E}_{ic}^{\mathrm{fm}}$ from the calibrated $\boldsymbol{z}_0^c$ without resampling

end if

if ZSL **then**

  $\mathcal{E}_{ic}^{\mathrm{vae}} \leftarrow \big\|D^s(\boldsymbol{\mu}_c^t) - \boldsymbol{e}_i\big\|^2$ ▷ **DDI**, evidence 2, deterministic

  $\hat{y}_i \leftarrow \arg\min_{c\in\mathcal{U}} \mathcal{E}_{ic}^{\mathrm{fm}} + \lambda\,\mathcal{E}_{ic}^{\mathrm{vae}}$ ▷ fused ranking

else

  $r_i \leftarrow \min_{c\in\mathcal{S}} \mathcal{E}_{ic}^{\mathrm{fm}} \,/\, \min_{c\in\mathcal{U}} \mathcal{E}_{ic}^{\mathrm{fm}}$

  $\hat{y}_i \leftarrow \begin{cases} \arg\min_{c\in\mathcal{U}} \mathcal{E}_{ic}^{\mathrm{fm}}, & r_i > \gamma \\ \arg\min_{c\in\mathcal{S}} \mathcal{E}_{ic}^{\mathrm{fm}}, & r_i \le \gamma \end{cases}$ ▷ gated ranking

end if

return $\{\hat{y}_i\}_{i=1}^{B}$

Table 7: Implementation and hyper-parameter details.

Training	
Optimizer	AdamW, lr 1×10^{-4}, weight decay 0.01
Iterations	1200 (first 1000 alignment-only)
Batch size	64
Latent dimension D	768
Text embedding dimension	1536
Velocity predictor v_θ	DiT, 1 layer, 12 heads, MLP ratio 4
Flow step ratio	0.5
Low-rank size r	4 (NTU-60/120), 1 (PKU-MMD)
Shape alignment weight λ_{shape}	0.1
Velocity contrastive weight	$0.1 \to 0$ (linear)
GZSL gate threshold γ	0.75 (NTU-60/120), 0.45 (PKU-MMD)
Text description top-k	2 (NTU-60/120), 3 (PKU-MMD)
Inference	
Fusion weight λ	0.4 (55/5), 0.1 (48/12), 0.15 (110/10), 0.45 (96/24), 0.1 (PKU-MMD)
Vote temperature τ	1
Calibration budget K	60
TSC	shrinkage only, EMA and adaptive strength off

evidence of DDI, which contributes only 1.8% of the total. The dominant costs, the semantic encoder (33.1%) and the flow-matching scoring (31.3%), are shared with the baseline and follow from the pipeline rather than from our components. In contrast to test-time approaches that optimize prototypes by gradient descent or rely on self-pretrained backbones, all extra computation of LoAD reduces to a few closed-form passes on a frozen model, which we consider a favorable trade-off between accuracy and efficiency.

Table 8: Computational cost of LoAD against the Flora pipeline, measured with $K = 12$ unseen classes.

	Flora	LoAD
Trainable parameters (M)	15.81	18.96
low-rank heads of LRCA	-	+3.15
parameters added by TDA	-	0
FLOPs per sample (G)	8.25	12.34
velocity-error re-score after TSC	-	3.86
reconstruction evidence of DDI	-	0.23
anchor statistics of TSC	-	≈ 0
Test-time gradient updates	-	0