Bridging the Skeleton-Text Modality Gap: Diffusion-Powered Modality Alignment for Zero-shot Skeleton-based Action Recognition

Supplementary Material

A. Limitations

A.1. Sensitivity to Noise Variation

As illustrated in Fig. 4, although our method achieves superior performance, its results exhibit some sensitivity to noise ϵ_{test} during inference. Recent studies [21] have suggested that predicting the initial state $\mathbf{z}_x$ during the reverse diffusion process yields more stable results compared to direct noise (ϵ) prediction, especially under varying noise conditions. As part of future work, we plan to explore this refinement to enhance the robustness against noise fluctuations.

Predicting $\mathbf{z}_x$. Our model exhibits somewhat sensitivity to noise during inference. To address this, we additionally experimented with predicting $\mathbf{z}_x$ instead of noise ϵ. As shown in Fig. 5, noise-induced fluctuation is reduced by $5\times$, with minimal performance drop and SOTA-level accuracy maintained.

B. Additional Discussions on Results

B.1. Results on More Complex Datasets

We provide additional results on Kinetics-200 and Kinetics-400 datasets [28] in Table 7 and Table 8. For consistency, we use the same skeleton encoder and same text prompts as PURLS [79]. Notably, despite using only a *single* text prompt per action, our method achieves state-of-the-art performance across all data splits, outperforming prior approaches that leverage multiple text prompts.

B.2. More Comparison with BSZSL

We did additional comparison with BSZSL [40] that utilizes both text and RGB modalities. Without relying on RGB input, our TDSM in the Table 9 outperforms BSZSL in 3 out of 4 splits across NTU-60 and NTU-120 datasets.

B.3. Analysis of Split Settings

Table 2 presents the average performance of our model across split 1, split 2, and split 3 on the SMIE [77] benchmark. For a more detailed analysis, Table 10 reports the performance for each individual split. Notably, in the NTU-60 55/5 split, our TDSM achieves the highest performance for split 2. The unseen classes in this split—"*wear a shoe*", "*put on a hat/cap*", "*kicking something*", "*nausea or vomiting condition*", and "*kicking other person*"—exhibit clear and distinct motion patterns. For example, "*wear a shoe*" involves downward torso motion, "*put on a hat/cap*" features upward hand movements, "*kicking something*" emphasizes significant leg activity, "*nausea or vomiting condition*" depicts upper body contraction, and "*kicking other person*" is unique as it involves two skeletons interacting. These distinct characteristics make our TDSM easier to distinguish the classes, leading to higher performance. Fig. 6 illustrates this trend through the confusion matrix and per-class accuracy visualization, highlighting the clear separability of these actions.

In contrast, our TDSM shows relatively lower performance for the split 1 in the PKU-MMD 46/5 split, although the split 1 contains fewer unseen classes ("*falling*", "*make a phone call/answer phone*", "*put on a hat/cap*", "*taking a selfie*", and "*wear on glasses*"). Except for "*falling*", the remaining classes involve similar upward hand movements and interactions with objects (e.g., phones, hats, glasses) that are not explicitly visible in skeleton data. This lack of contextual information makes it significantly harder to distinguish these actions, resulting in degraded performance. As visualized in Fig. 7, the confusion matrix and per-class accuracy further reveal the challenge of separating actions with overlapping motion patterns, emphasizing the limitations of skeleton-only data when distinguishing semantically similar actions. These observations underscore the importance of distinct motion patterns in unseen classes for robust zero-shot recognition.

B.4. Potential Training-Inference Mismatch

Our TDSM adopts an one-step inference framework, where both training and inference are consistently performed with the same total number of timesteps T. Empirically, we found that performing one-step inference at $t_{\text{test}} = T/2$ (e.g., $t_{\text{test}} = 50$ when $T = 100$) provides the best trade-off between noise and structure. So, no distributional mismatch exists b/w training and inference in our setting.

B.5. More Explanation about Text Feature

Compared to PURLS [79] where local textual features are obtained from six separate body-part-specific descriptions, our TDSM extracts both global and local features from a single unified sentence, yielding and $\mathbf{z}_g$ and $\mathbf{z}_l$ which are described in details.

C. Discussion on ZSAR Methods

The key contribution of our work lies not in each individual component (e.g., DiT architecture, loss function but in proposing a new framework for ZSAR that effectively bridges the cross-modality gap between skeleton and

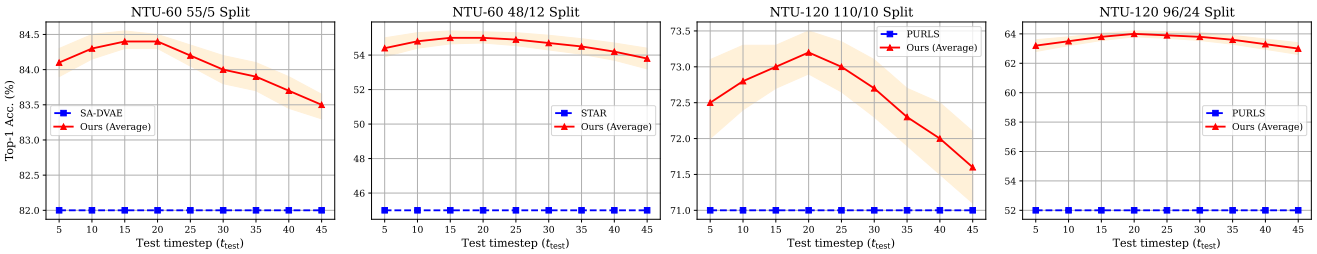

Figure 5. Effect of varying inference timesteps t_{test} across multiple datasets. Each plot shows the top-1 accuracy trend on the NTU-60 and NTU-120 datasets under different splits. The solid red line represents the average accuracy of our method, with the shaded orange area indicating the variation in accuracy across 10 different random Gaussian noise instances. Dashed blue line corresponds to the second-best method in each benchmark.

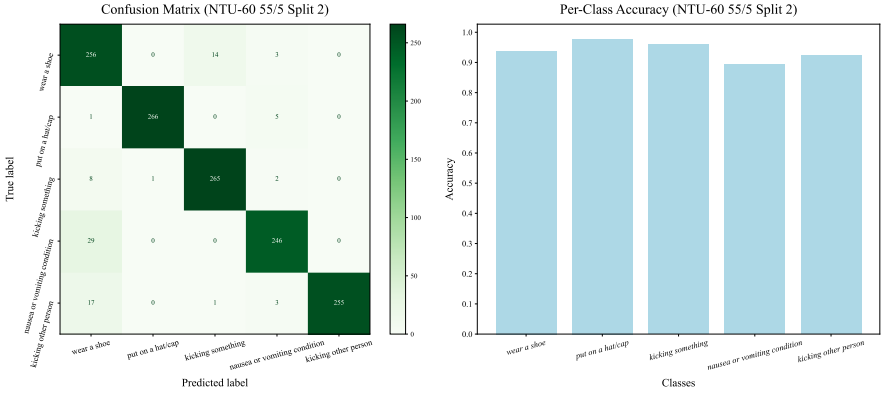

Figure 6. Confusion matrix and per-class top-1 accuracy visualization for NTU-60 55/5 Split 2.

Methods	Kinetics-200 (Acc, %)			
	180/20 split	160/40 split	140/60 split	120/80 split
ReViSE [26]	24.95	13.28	8.14	6.23
DeViSE [18]	22.22	12.32	7.97	5.65
PURLS [79] (1 text)	25.96	15.85	10.23	7.77
PURLS [79] (7 text)	32.22	22.56	12.01	11.75
TDSM (1 text)	**38.18**	**24.43**	**15.28**	**13.09**

Table 7. Top-1 accuracy results of TDSM evaluated on the Kinetics-200 dataset under the PURLS [79] benchmark.

Methods	Kinetics-400 (Acc, %)			
	360/40 split	320/80 split	300/100 split	280/120 split
ReViSE [26]	20.84	11.82	9.49	8.23
DeViSE [18]	18.37	10.23	9.47	8.34
PURLS [79] (1 text)	22.50	15.08	11.44	11.03
PURLS [79] (7 text)	34.51	24.32	16.99	14.28
TDSM (1 text)	**38.92**	**26.24**	**18.45**	**16.10**

Table 8. Top-1 accuracy results of TDSM evaluated on the Kinetics-400 dataset under the PURLS [79] benchmark.

text. Previous VAE-based or contrastive learning(CL)-based methods attempt *direct* alignment between skeleton and text latents, but the inherent large modality gap limits their effectiveness (Sec. 2.1). To address this, we use a diffusion process—already shown to be powerful in image-text alignment—and adapt it for a discriminative zero-

Methods	Modality		NTU-60 (Acc, %)		NTU-120 (Acc, %)	
	Text	RGB	55/5 split	48/12 split	110/10 split	96/24 split
BSZSL [40]	✓	✓	83.04	52.96	**77.69**	56.12
TDSM	✓		**86.49**	**56.03**	74.15	**65.06**

Table 9. Top-1 accuracy results of BSZSL evaluated on the SynSE and PURLS benchmarks for the NTU-60 and NTU-120 datasets.

TDSM (Ours)	NTU-60 (Acc, %)	NTU-120 (Acc, %)	PKU-MMD (Acc, %)
	55/5 split	110/10 split	46/5 split
Split 1	87.97	74.45	57.40 (Fig. 7)
Split 2	96.06 (Fig. 6)	63.91	76.92
Split 3	82.60	70.04	77.97
Average	88.88	69.47	70.76

Table 10. Top-1 accuracy results of our TDSM evaluated on the NTU-60, NTU-120, and PKU-MMD datasets under the SMIE [77] benchmark.

shot action recognition task. The implication of our DM-based TDSM is very meaningful as Table 11, which has brought out significant performance improvement with 2.36 to 13.05%-point.

C.1. Difference against Previous ZSAR Methods

As illustrated in Table 11, previous VAE-based and CL-based methods rely on *explicit point-wise alignment*, min-

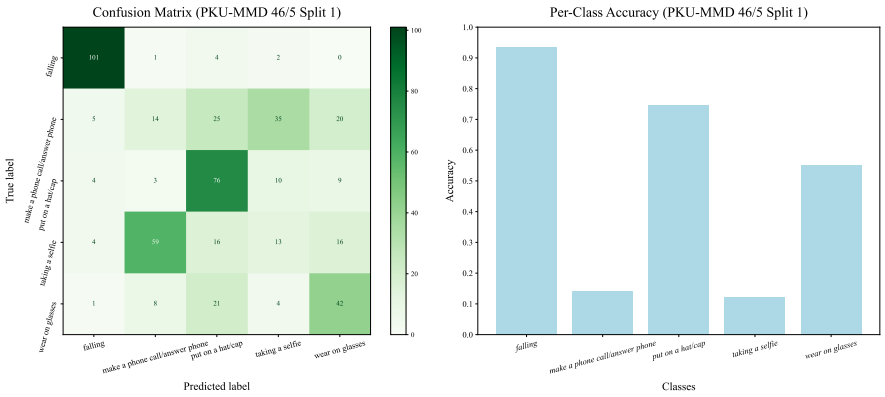

Figure 7. Confusion matrix and per-class top-1 accuracy visualization for PKU-MMD 46/5 Split 1.

imizing cross-reconstruction error or feature distances directly between skeleton and text features. But, our TDSM aligns the two modalities implicitly by learning to denoise a noisy skeleton feature in a single reverse diffusion step conditioned on a text feature.

C.2. Why Diffusion is Effective for ZSAR?

Diffusion models are known for their strong cross-modal alignment capabilities, enabled through conditioning mechanisms that integrate signals. Our TDSM leverages this property using an one-step reverse diffusion process conditioned on a text embedding, to denoise a skeleton feature. We believed this property of diffusion (its ability to integrate semantic guidance during denoising) would be particularly effective for ZSAR, where bridging modality gaps is critical. To the best of our knowledge, our TDSM is the first to apply diffusion in this discriminative alignment setting for ZSAR, validating its effectiveness across multiple benchmarks.

In comparison with the previous work [6, 17, 19, 24, 25, 66], they are based on diffusion models and have generation tasks while our TDSM utilizes the property of diffusion model's strong cross-modality alignment for discriminative tasks, not for generation tasks. Note that [17] predicts actions by generating visual representations in an iterative sampling process, while our TDSM utilizes a diffusion model in a single-step inference without generating any feature for action classification.

D. Detailed Structure of the Diffusion Transformer

D.1. About the Diffusion Transformer Design

Note that our main contribution does not lie in the design of new components, but the first diffusion-based framework that is built upon DiT [48] and MMDiT [15] that have been well-validated for cross-modality alignment. Unlike the original DiT, we replace the class label embedding with $\mathbf{z}_g$ for semantic conditioning. Also shown in Table 12, we also experimented with a U-Net backbone [53], but found DiT to perform better in our setting.

D.2. Diffusion Transformer Architecture

The Diffusion Transformer $\mathcal{T}_{\text{diff}}$ takes $\mathbf{z}_{x,t}$, $\mathbf{z}_g$, $\mathbf{z}_l$, and t as inputs (Fig. 2 in the main paper). These inputs are embedded into corresponding feature representations $\mathbf{f}_{x,t}$, $\mathbf{f}_c$, and $\mathbf{f}_l$ as follows:

$$\begin{aligned} \mathbf{f}_{x,t} &= \mathsf{Linear}(\mathbf{z}_{x,t}) + \mathsf{PE}_x, \\ \mathbf{f}_c &= \mathsf{Linear}(\mathsf{TE}_t) + \mathsf{Linear}(\mathbf{z}_g), \\ \mathbf{f}_l &= \mathsf{Linear}(\mathbf{z}_l) + \mathsf{PE}_l, \end{aligned} \tag{16}$$

where PE_x and PE_l are positional embeddings applied to the feature maps, capturing spatial positional information, while TE_t is a timestep embedding [60] that maps the scalar t to a higher-dimensional space. The embedded features $\mathbf{f}_{x,t}$, $\mathbf{f}_c$, and $\mathbf{f}_l$ are then passed through B CrossDiT Blocks, followed by a Layer Normalization (LN) and a final Linear layer to predict the noise $\hat{\boldsymbol{\epsilon}} \in \mathbb{R}^{M_x \times C}$.

D.3. CrossDiT Block

The CrossDiT Block facilitates interaction between skeleton and text features, enhancing fusion through effective feature modulation [49, 65] and multi-head self-attention [61]. Fig. 8 shows a detail structure of our CrossDiT Block. Built upon the DiTs architecture [15, 48], it leverages modulation techniques and self-attention mechanisms to efficiently capture the dependencies across these modalities. The skeleton feature $\mathbf{f}_x$ and local text feature $\mathbf{f}_l$ are first modulated separately using $\mathsf{Scale\text{-}Shift}$ and Scale operations as:

$$\begin{aligned} &[\boldsymbol{\alpha}_x \mid \boldsymbol{\beta}_x \mid \boldsymbol{\gamma}_x \mid \boldsymbol{\alpha}_l \mid \boldsymbol{\beta}_l \mid \boldsymbol{\gamma}_l] = \mathsf{Linear}(\mathbf{f}_c), \\ &\mathsf{Scale\text{-}Shift}: \ \mathbf{f}_i \leftarrow (1+\boldsymbol{\gamma}_i) \odot \mathbf{f}_i + \boldsymbol{\beta}_i, \\ &\mathsf{Scale}: \ \mathbf{f}_i \leftarrow \boldsymbol{\alpha}_i \odot \mathbf{f}_i, \end{aligned} \tag{17}$$

Methods	Characteristics	Limitations
VAE-based	Reconstructs skeleton-text feature pairs via **cross-reconstruction,** recovering skeleton features from text and vice versa	Modality gap due to direct alignment
CL-based	Aligns skeleton and text features by **minimizing feature distance** through contrastive learning	
TDSM (Ours)	Denoises skeleton latents (i.e., estimates added noise in the forward diffusion) using **reverse diffusion**, conditioned on text embeddings, to naturally align both modalities in a **unified latent space**	Noise-sensitive performance

Table 11. Comparison with ours TDSM with existing ZSAR methods.

Backbone	NTU-60 (Acc, %)		NTU-120 (Acc, %)	
	55/5 split	48/12 split	110/10 split	96/24 split
U-Net [53]	82.40	51.12	70.03	59.77
DiT (**TDSM**)	**86.49**	**56.03**	**74.15**	**65.06**

Table 12. Comparison with ours TDSM with existing ZSAR methods.

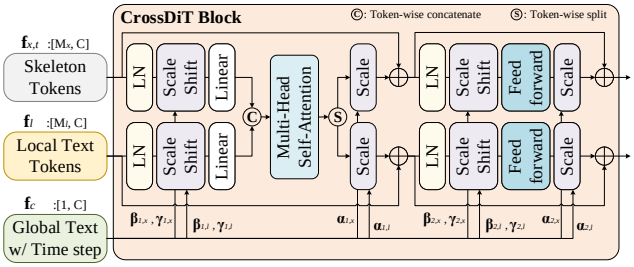

Figure 8. A detail structure of our CrossDiT Block.

where $i \in \{x, l\}$ denotes the skeleton or local text feature, respectively. The parameters $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$ are conditioned on the global text feature $\mathbf{z}_g$ and timestep t, allowing the block to modulate feature representations effectively. Also, we compute query, key, and value matrices for both skeleton and local text features separately:

$$[\mathbf{q}_i \mid \mathbf{k}_i \mid \mathbf{v}_i] = \mathsf{Linear}(\mathbf{f}_i). \tag{18}$$

These matrices are token-wise concatenated and fed into a multi-head self-attention module, followed by a split to retain token-specific information as:

$$[\mathbf{f}_x \mid \mathbf{f}_l] \leftarrow \mathsf{SoftMax}\left([\mathbf{q}_x \mid \mathbf{q}_l]\,[\mathbf{k}_x \mid \mathbf{k}_l]^{\mathsf{T}}\right)[\mathbf{v}_x \mid \mathbf{v}_l]\,. \tag{19}$$

By leveraging the attention from skeleton, timestep, and text features, the CrossDiT Block ensures efficient interaction between modalities, promoting the skeleton-text fusion for discriminative feature learning and improved generalization to unseen actions.

E. Implementation Details

Table 13 provides a detailed summary of the variables used in TDSM. We utilized $B = 12$ CrossDiT Blocks, each containing a multi-head self-attention module with 12 heads.

	Module	Output Shape
$\mathbf{X}$		$T \times V \times M \times 3$
$\mathbf{z}_x$	$\mathcal{E}_x$	$M_x \times 256$
$\mathbf{f}_{x,t}$	$\mathbf{z}_x$ Embed	$M_x \times 768$
$\mathbf{z}_g$	$\mathcal{E}_d$	1×1024
$\mathbf{z}_l$		$M_l \times 1024$
$\mathbf{f}_c$	t Embed $\mathbf{z}_g$ Embed	1×768
$\mathbf{f}_l$	$\mathbf{z}_l$ Embed	$M_l \times 768$
$\boldsymbol{\epsilon}, \hat{\boldsymbol{\epsilon}}$		$M_x \times 256$

Table 13. The details of feature shape.

All feature dimensions were set to $C = 768$. The local text features contained $M_l = 35$ tokens, while the skeleton features were represented by a single token $M_x = 1$. To ensure reproducibility, the random seed was fixed at 2,025 throughout all experiments. Skeleton features ($\mathbf{z}_x$) are extracted using skeleton encoder (Shift-GCN [9] or ST-GCN [71]), resulting in a channel dimension of 256. For text features, two descriptions per action are encoded using the CLIP [27, 51] text encoder, producing features with a channel dimension of 1,024. These features are concatenated along the channel dimension to form a unified text representation.

Fair comparison. For the SynSE and PURLS settings, we utilize the same encoders as prior works to maintain consistency. We also encode $\mathbf{X}$ into $\mathbf{z}_x$ with $M_x = 1$ to avoid any advantage from higher-resolution features (e.g., $M_x = T \times V$), again to ensure fair evaluation. These are a common practice in ZSAR task. For fair comparison, we used the same text prompts employed in existing works. When publicly available text prompts were provided, we used them as they were and did not heavily modify or augment them. For datasets without text descriptions (e.g., PKU-MMD [39]), we used GPT-4 [1] to generate single description per action, ensuring consistency with the existing text styles.

Hyper-parameter. We tuned hyper-parameters extensively on the NTU-60 SynSE benchmark and then applied the same settings to all other datasets. Our method still achieved SOTA results.

F. Additional Related Work

F.1. Skeleton-based Action Recognition

Traditional skeleton-based action recognition assumes fully annotated training and test datasets, in contrast to other skeleton-based action recognition methods under zero-shot settings which aim to recognize unseen classes without explicit training samples. Early methods [37, 41, 74, 80] employed RNN-based models to capture the temporal dynamics of skeleton sequences. Subsequent studies [3, 13, 29, 70] explored CNN-based approaches, transforming skeleton data into pseudo-images. Recent advancements leverage graph convolutional networks (GCNs) [7, 10, 33, 34, 43, 68, 75, 78] to effectively represent the graph structures of skeletons, comprising joints and bones. ST-GCN [71] introduced graph convolutions along the skeletal axis combined with 1D temporal convolutions to capture motion over time. Shift-GCN [9] improved computational efficiency by implementing shift graph convolutions. Building on these methods, transformer-based models [12, 14, 46, 62, 76] have been proposed to address the limited receptive field of GCNs by capturing global skeletal-temporal dependencies. In this work, we adopt ST-GCN [71] and Shift-GCN [9] to extract skeletal-temporal representations from skeleton data, transforming input skeleton sequences into a latent space for further processing in the proposed framework.