Reconstruction-Anchored Diffusion Model for Text-to-Motion Generation — Supplementary Material —

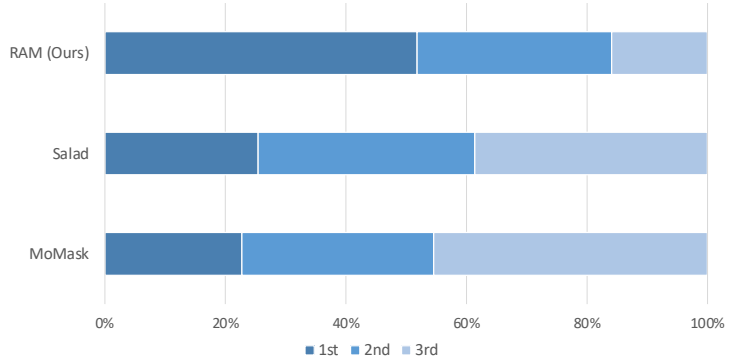

Fig. 1: User study results on the HumanML3D test set comparing RAM with state-of-the-art methods. We conducted a perceptual study to evaluate human preferences based on a holistic assessment of two key dimensions: motion quality and semantic alignment. Participants were instructed to rank the generated motions from different methods by jointly considering these factors. The results indicate that RAM achieves the best overall performance, demonstrating its superior capability in generating motions that are both realistic and semantically accurate.

A User Study

To further validate the effectiveness of our method from a perceptual perspective, we conduct a user study comparing three methods: MoMask [2], Salad [3], and our proposed RAM. We randomly select 30 text prompts from the test set and generate one motion sample per method for each prompt, resulting in 90 motion samples in total.

We recruit 15 participants, each holding at least a bachelor's degree. Each participant is asked to evaluate all 30 text prompts. For each prompt, the participant views the three generated motions (presented in random order to avoid bias) and ranks them from best to worst based on two criteria: (1) text-motion alignment, i.e., how well the generated motion matches the semantic content of the text, and (2) motion quality, i.e., the naturalness and realism of the generated motion. Participants are instructed to consider both criteria comprehensively when providing their rankings.

The results of the user study are presented in Fig. 1, which shows the distribution of rankings across all evaluations. Our method achieves the best overall performance, securing the first rank in over 50% of the cases. This result demonstrates that our approach generates motions that are not only better aligned with the input text but also exhibit superior quality compared to MoMask and Salad.

B Inference Efficiency

Table 1: Experiments on inference efficiency. We compare the Average Inference Time per Sentence (AITS) and generation quality of RAM (20 steps) against MDM (50 steps) with varying REG configurations. The results demonstrate that RAM achieves higher efficiency (lower AITS) than MDM even when REG is fully enabled. Moreover, applying REG only in the early denoising steps yields significant improvements in FID, confirming that correcting errors early in the diffusion process is critical for mitigating error propagation and enhancing motion realism.

Method	enable REG at step t	AITS↓	FID↓	R-Precision Top 1	Top 2	Top 3
MDM (50steps)	None	0.490	$0.398^{\pm.010}$	$0.456^{\pm.002}$	$0.646^{\pm.003}$	$0.752^{\pm.002}$
RAM (20steps)	None	0.226	$0.132^{\pm.005}$	$0.561^{\pm.002}$	$0.752^{\pm.002}$	$0.838^{\pm.002}$
	[46,44]	0.235	$0.075^{\pm.003}$	$0.561^{\pm.002}$	$0.750^{\pm.002}$	$0.838^{\pm.001}$
	[46, 44, 41, 39]	0.261	$0.046^{\pm.002}$	$0.561^{\pm.003}$	$0.750^{\pm.002}$	$0.838^{\pm.002}$
	[46, 44, 41, 39, 36, 34]	0.284	$0.038^{\pm.002}$	$0.560^{\pm.003}$	$0.750^{\pm.001}$	$0.838^{\pm.002}$
	All except the inital step	0.398	$0.032^{\pm.002}$	$0.561^{\pm.003}$	$0.751^{\pm.002}$	$0.839^{\pm.002}$

The proposed Reconstructive Error Guidance (REG) introduces an additional reconstruction step for previous predictions during inference, which increases inference time. However, experiments show that RAM requires substantially fewer inference steps than most commonly used motion diffusion models. As discussed in Sec. 4.2 of the main paper, training uses $T = 50$ diffusion steps. For efficient inference, we automatically select 20 denoising steps by linearly spacing them within the interval $[0, \ldots, T-1]$, resulting in the following indices:

$$t = [49,\ 46,\ 44,\ 41,\ 39,\ 36,\ 34,\ 31,\ 28,\ 26,\ 23,\ 21,\ 18,\ 15,\ 13,\ 10,\ 8,\ 5,\ 3,\ 0].$$

We compare MDM (50steps) and several configurations where REG is applied at different steps, reporting both their average inference time per sentence (AITS) [1] and the resulting generation quality. AITS is calculated on the HumanML3D test set by setting the batch size to 1 and excluding model and dataset loading time. Note that MDM (50steps) is an improved variant of the original MDM, offering higher inference efficiency and better generation results compared to those reported in the original paper [5].

Table 2: Effect of loss parameters β and τ. We study the gradient control parameter β in latent alignment and the temperature τ in self-regularization. A small β (e.g., 0.01) yields the best performance by preserving essential motion information, while unrestricted flow ($\beta = 1$) degrades results. For τ, we find that the model is relatively robust to this parameter, with $\tau = 1.0$ achieving the best performance.

β	τ	FID↓	R-Precision↑ Top 1	Top 2	Top 3	MM D↓	Diversity
1.00	1.0	$0.278^{\pm.008}$	$0.498^{\pm.002}$	$0.696^{\pm.002}$	$0.797^{\pm.002}$	$3.001^{\pm.008}$	$9.777^{\pm.075}$
0.10	1.0	$0.132^{\pm.003}$	$0.530^{\pm.001}$	$0.728^{\pm.003}$	$0.823^{\pm.003}$	$2.847^{\pm.007}$	$9.669^{\pm.067}$
0.01	1.0	$0.032^{\pm.002}$	$0.561^{\pm.003}$	$0.751^{\pm.002}$	$0.839^{\pm.002}$	$2.716^{\pm.007}$	$9.487^{\pm.084}$
0.00	1.0	$0.045^{\pm.003}$	$0.563^{\pm.002}$	$0.751^{\pm.003}$	$0.839^{\pm.002}$	$2.707^{\pm.006}$	$9.438^{\pm.053}$
0.01	2.0	$0.055^{\pm.003}$	$0.561^{\pm.002}$	$0.753^{\pm.003}$	$0.842^{\pm.002}$	$2.704^{\pm.007}$	$9.537^{\pm.078}$
0.01	1.0	$0.032^{\pm.002}$	$0.561^{\pm.003}$	$0.751^{\pm.002}$	$0.839^{\pm.002}$	$2.716^{\pm.007}$	$9.487^{\pm.084}$
0.01	0.5	$0.039^{\pm.003}$	$0.560^{\pm.002}$	$0.751^{\pm.002}$	$0.841^{\pm.002}$	$2.721^{\pm.008}$	$9.533^{\pm.076}$

The results summarized in Tab. 1 show that, due to fewer inference steps, RAM achieves significantly higher efficiency even when REG is enabled at every step (except the initial step , which lacks a previous prediction). Notably, enabling REG only in the early denoising steps already leads to a marked improvement in FID. This supports our claim in the introduction that early denoising steps—responsible for recovering motion from nearly pure noise—are particularly prone to generating error patterns, and thus benefit most from the application of REG.

To further characterize the overhead, we measure the per-step cost of each guidance configuration. At a batch size of 32, the per-step time ratio of *baseline*:*CFG*:*REG*:*CFG+REG* is approximately 1:1.95:2.18:3.13, i.e. REG is essentially at parity with CFG per step. Combined with the early-window analysis above, applying REG only to the first 6 steps adds just 26% inference time (AITS 0.226→0.284 s) while improving FID by 71% (0.132→0.038), offering a favorable quality–efficiency trade-off.

C Hyperparameter Analysis

In this section, we analyze the impact of key hyperparameters to verify our design choices and understand their influence on model performance. Our investigation covers three main aspects: (1) *Internal loss parameters*—evaluating β and τ, where β controls how strongly the latent alignment loss L_{latent} updates the motion encoder E_m, and τ shapes the geometry of the motion latent space through the sharpness of similarities in self-regularization; (2) *Loss weights*—assessing the balance between motion-centric latent alignment and self-regularization; and (3) *Latent dimension*—exploring the trade-off between representation capacity and model compactness.

Internal Loss Parameters. Tab. 2 illustrates how the internal parameters β and τ influence generation quality. β controls the gradient flow from the latent alignment loss L_{latent} to the motion encoder E_m, with β equal to 0 fully

blocking the gradient and β equal to 1 allowing unrestricted gradient flow. Our results show that allowing equal proximity between text and motion latents (that is, when β is equal to 1) is suboptimal, as this alignment comes at the expense of motion information in the motion latents. In fact, reducing the gradient flow to E_m improves performance, with the best results achieved when β is set to 0.01. We attribute this to the constant evolution of the motion latent space during training, which increases the difficulty of latent alignment. Moreover, enforcing strong alignment gradients on E_m can interfere with learning a motion-discriminative latent space while it is still evolving, making the alignment objective harder to satisfy. By setting β to 0.01, we ease the alignment process while preserving essential motion information in the latent space.

Another parameter τ determines the sharpness of similarity in L_{sr}. A smaller τ produces sharper similarities, pushing motion latents farther apart; if too extreme, this can distort the structure of the motion manifold. In contrast, a larger τ smooths the similarity, relaxing the constraints between latents, but may reduce gains in semantic resolution. Empirically, we found $\tau = 1$ offers a desirable balance.

Table 3: Effect of loss weights. We evaluate the impact of varying the weights for the self-regularization (w_{sr}) and latent alignment (w_{latent}) objectives. The quantitative results show that RAM maintains stable performance across different configurations, indicating that the model is robust to variations in these hyperparameters.

w_{latent}	w_{sr}	FID↓	R-Precision↑ Top 1	Top 2	Top 3	MM Dist↓	Diversity
1.0	1.0	$0.037^{\pm.002}$	$0.563^{\pm.003}$	$0.755^{\pm.002}$	$0.843^{\pm.002}$	$2.693^{\pm.008}$	$9.496^{\pm.094}$
1.0	0.5	$0.055^{\pm.003}$	$0.560^{\pm.003}$	$0.751^{\pm.003}$	$0.841^{\pm.002}$	$2.703^{\pm.009}$	$9.482^{\pm.080}$
1.0	0.1	$0.063^{\pm.004}$	$0.555^{\pm.004}$	$0.747^{\pm.002}$	$0.837^{\pm.002}$	$2.729^{\pm.006}$	$9.532^{\pm.074}$
1.0	0.0	$0.109^{\pm.005}$	$0.533^{\pm.003}$	$0.722^{\pm.002}$	$0.815^{\pm.002}$	$2.859^{\pm.009}$	$9.508^{\pm.084}$
0.5	1.0	$0.032^{\pm.002}$	$0.561^{\pm.003}$	$0.751^{\pm.002}$	$0.839^{\pm.002}$	$2.716^{\pm.007}$	$9.487^{\pm.084}$
0.1	1.0	$0.056^{\pm.002}$	$0.534^{\pm.003}$	$0.730^{\pm.002}$	$0.822^{\pm.001}$	$2.839^{\pm.007}$	$9.465^{\pm.062}$
0.0	1.0	$0.422^{\pm.016}$	$0.476^{\pm.003}$	$0.673^{\pm.003}$	$0.778^{\pm.002}$	$3.134^{\pm.010}$	$9.451^{\pm.075}$

Loss Weights. We investigate the impact of the weights w_{latent} and w_{sr}, which correspond to motion-centric latent alignment and self-regularization, respectively. The results are presented in Tab. 3. It can be observed that setting either weight to zero results in a significant performance drop. However, as long as both weights are nonzero, changes in their values have only a minor effect on performance. These findings highlight the importance of each objective component and demonstrate RAM's robustness to variations in weight assignment.

Encoder Latent Dimension. The dimensionality of the encoder's latent space, d_E—which determines the size of the motion and text latents—is a key hyperparameter in our model. A larger d_E can capture more intricate details but increases the model size and may raise the risk of overfitting, while a smaller d_E

Table 4: Effect of encoder latent dimension d_E. The results indicate that $d_E = 256$ yields the optimal balance, achieving the lowest FID score. Furthermore, the model exhibits robustness to dimension variations, with even a reduced dimension of 128 maintaining competitive performance, highlighting the compactness of the learned motion latent space.

d_E	FID↓	R-Precision↑ Top 1	Top 2	Top 3	MM Dist↓	Diversity
128	$0.043^{\pm.002}$	$0.546^{\pm.002}$	$0.739^{\pm.002}$	$0.829^{\pm.002}$	$2.767^{\pm.008}$	$9.509^{\pm.088}$
192	$0.050^{\pm.003}$	$0.555^{\pm.002}$	$0.748^{\pm.003}$	$0.838^{\pm.002}$	$2.725^{\pm.008}$	$9.545^{\pm.072}$
256	$0.032^{\pm.002}$	$0.561^{\pm.003}$	$0.751^{\pm.002}$	$0.839^{\pm.002}$	$2.716^{\pm.007}$	$9.487^{\pm.084}$
384	$0.049^{\pm.002}$	$0.555^{\pm.002}$	$0.748^{\pm.002}$	$0.836^{\pm.002}$	$2.746^{\pm.009}$	$9.618^{\pm.104}$
512	$0.064^{\pm.003}$	$0.558^{\pm.003}$	$0.753^{\pm.002}$	$0.842^{\pm.002}$	$2.724^{\pm.007}$	$9.583^{\pm.085}$

may result in information loss. In our main experiments, we set d_E to 256. Here, we further explore how varying d_E influences RAM's performance.

As shown in Tab. 4, altering d_E leads to only minor fluctuations in performance, indicating that RAM is relatively robust to this hyperparameter. Interestingly, even when d_E is halved to 128, RAM's performance only decreases slightly. This suggests that the learned latent space is highly compact.

D Implementation Details of Latent Space Training Strategies

This section provides the detailed implementation configurations for the comprehensive comparison of latent space training strategies discussed in Sec. 4.4 of the main paper.

Network Architecture and Hyperparameters. To ensure a strictly fair comparison, all evaluated variants (Variants A–E and our proposed method) share identical foundational configurations, differing only in their training objectives applied within the latent space. Specifically, all models employ the exact same diffusion-based decoder and maintain equivalent model architectures with the same number of parameters. Furthermore, the optimization hyperparameters, including the learning rate, batch size, and the total number of training steps, are kept strictly consistent across all experiments.

Training Objectives. Regarding the loss functions, for any loss weights that are not explicitly specified in the variant descriptions in the main text, the default value is set to 1.0. Additionally, since our framework does not employ a Variational Autoencoder (VAE) structure, none of the compared variants utilize the Kullback-Leibler (KL) divergence loss. The specific loss weights for bidirectional latent alignment and cross-modal contrastive learning (*e.g.*, temperature $\tau = 0.1$) follow the exact settings described in the main paper to faithfully reproduce the representative methods (such as TEMOS and TMR).

Inference Settings. At inference time, to isolate the impact of the latent space representations on generation quality, all methods apply standard classifier-free

Table 5: Leave-one-out ablation on HumanML3D. Starting from the full RAM, we remove a single component at a time.

	Full	w/o L_{latent}	w/o L_{sr}	w/o REG
FID↓	$\mathbf{0.032}^{\pm.002}$	$0.424^{\pm.016}$	$0.064^{\pm.003}$	$0.132^{\pm.005}$
R-Precision Top 1↑	$\mathbf{0.561}^{\pm.003}$	$0.476^{\pm.003}$	$0.536^{\pm.003}$	$\mathbf{0.561}^{\pm.002}$

guidance (CFG). To ensure fairness in evaluating the baseline capabilities of the latent spaces, REG is not employed for any of the variants during this specific evaluation.

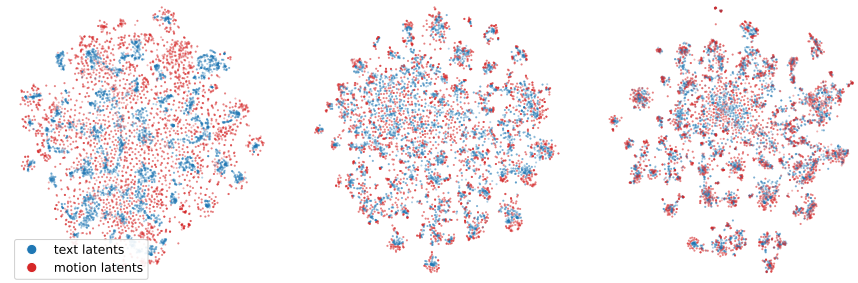

Fig. 2: Joint t-SNE of paired text and motion latents on HumanML3D.

Visualizing the Motion-Centric Latent. To directly verify that RAM learns a motion-centric latent space, we apply joint t-SNE to paired text and motion latents on the HumanML3D test set, as shown in Fig. 2. The two-stream baseline (Variant A) forms compact text clusters but a diffuse motion distribution, revealing a modality gap. The contrastive variant (Variant E, i.e. TMR [4]) aligns the two distributions and exposes cluster-level structure. RAM yields the sharpest clusters with the tightest within-cluster interleaving of paired text and motion latents, corroborated by the per-panel pairwise cosine similarity. This provides direct evidence that our motion-centric alignment preserves motion structure while pulling the text condition onto the motion manifold.

E Leave-One-Out Ablation

To complement the incremental ablation in Table 3 of the main paper, we additionally report a leave-one-out study in Tab. 5, where each component is removed in isolation from the full RAM. Removing L_{latent} causes the most severe degradation (FID 0.032→0.424), as the text and motion latents are no longer aligned. Removing L_{sr} degrades FID to 0.064 and Top-1 R-Precision to 0.536, confirming

that self-regularization is essential for a discriminative motion latent. Removing REG raises FID to 0.132 while leaving R-Precision unchanged, consistent with REG primarily improving motion realism rather than semantic alignment. None of the reduced variants approaches the full model, demonstrating that the components are complementary rather than redundant.

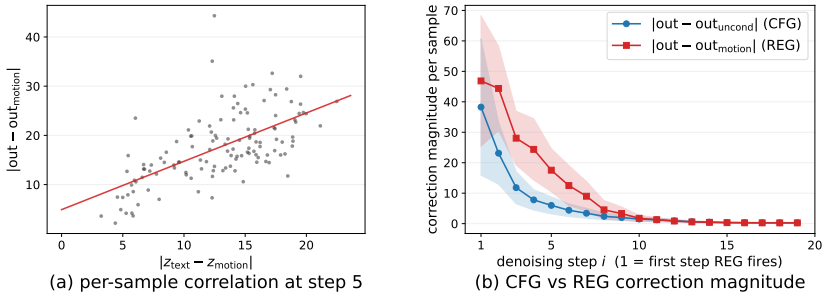

Fig. 3: REG: per-sample scatter (a), per-step magnitude (b).

F Empirical Analysis of REG's Correction

Fig. 3(a) examines REG at a single denoising step (B=128). Viewing the text condition as an anchor in the motion latent space (Sec. 3.2 of the main paper), the distance from the prior reconstruction latent to the text anchor serves as a proxy for the magnitude of the previous step's error. We find that this distance correlates positively with the magnitude of the REG correction (median Pearson $r\approx0.62$): the more erroneous the previous prediction, the stronger the correction REG applies. REG thus self-scales to where the model is wrong, rather than applying a uniform push.

References

1. Chen, X., Jiang, B., Liu, W., Huang, Z., Fu, B., Chen, T., Yu, G.: Executing your commands via motion diffusion in latent space. In: Computer Vision and Pattern Recognition (CVPR). pp. 18000–18010 (2023)
2. Guo, C., Mu, Y., Javed, M.G., Wang, S., Cheng, L.: Momask: Generative masked modeling of 3d human motions. In: Computer Vision and Pattern Recognition (CVPR). pp. 1900–1910 (2024)
3. Hong, S., Kim, C., Yoon, S., Nam, J., Cha, S., Noh, J.: Salad: Skeleton-aware latent diffusion for text-driven motion generation and editing. In: Computer Vision and Pattern Recognition (CVPR). pp. 7158–7168 (2025)
4. Petrovich, M., Black, M.J., Varol, G.: Tmr: Text-to-motion retrieval using contrastive 3d human motion synthesis. In: International Conference on Computer Vision (ICCV). pp. 9488–9497 (2023)

5. Shafir, Y., Tevet, G., Kapon, R., Bermano, A.H.: Human motion diffusion as a generative prior. arXiv preprint arXiv:2303.01418 (2023)