PHE-DiT: Phase-aware Hierarchical Energy Conditioning for Flexible-Length Dance-to-Music Generation

Anonymous authors
Paper under double-blind review

Abstract

Dance-to-music generation seeks to synthesize music that is temporally synchronized with human dance movements across multiple scales, requiring global tempo consistency, beat-level correspondence, and onset-level precision. However, existing methods typically condition on coarse motion features without explicitly modeling the hierarchical rhythmic structure of dance, and are constrained to fixed-length generation. We propose PHE-DiT, a conditional flow matching framework with a cross-modal transformer backbone that addresses these limitations through three key designs. First, *phase-aware input fusion* encodes instantaneous phase extracted from dance motion via the Hilbert transform as sinusoidal features and injects them into the input latent sequence, providing an explicit beat-cycle clock signal complementary to Rotary Position Embedding. Second, *hierarchical split energy conditioning* decomposes motion energy into slow and fast components via Butterworth filtering and applies them to earlier and later transformer layers via cross-attention, respectively, achieving coarse-to-fine alignment from beat-level to onset-level. Third, variable-length sequence support with length-aware attention masking enables *flexible-length music generation* from short clips to full-length choreographies within a single unified model. Extensive experiments on AIST++ and TikTok benchmarks demonstrate that PHE-DiT achieves state-of-the-art performance in rhythmic alignment, onset precision, and music quality.

1 Introduction

Music and dance exhibit an intrinsic coupling through shared rhythmic structure. Human choreography is organized around periodic motion patterns, dynamic accents, and hierarchical energy transitions, whereas music encodes these dynamics through structured tempo, beat, and onset patterns. Dance-to-music (D2M) generation (Zhu et al., 2022a; Yu et al., 2023; Wang et al., 2025; Sun et al., 2025; Im et al., 2026) aims to exploit this coupling and automatically synthesize realistic and rhythmically coherent music conditioned on input dance movements. This task lies at the intersection of human motion understanding (Zhu et al., 2023; Jiang et al., 2023) and music generation (Lam et al., 2023; Copet et al., 2023), and has attracted growing research interest due to its broad applications in entertainment, video production, and automated content creation. Unlike generic audio synthesis(Van Den Oord et al., 2016; Kong et al., 2020b), D2M imposes strict cross-modal alignment constraints where the generated music must maintain global tempo consistency, exhibit beat-level correspondence with motion periodicity, and achieve onset-level precision with fine-grained movement transients. Specifically, at the global level, the generated music should maintain consistent tempo with the overall motion dynamics of the choreography. Besides, at an intermediate level, beat structures need to align with periodic body movements and rhythmic motion patterns. Moreover, at finer temporal scales, transient musical events such as onsets should correspond to rapid motion accents and energy peaks in the dance. Modeling these hierarchical dependencies is therefore essential for producing rhythmically coherent music conditioned on dance movements.

Existing D2M generation methods have been developed under several generative paradigms, including GAN-based frameworks (Zhu et al., 2022a), Transformer-based architectures (Di et al., 2021), and, more recently, diffusion-based models (Sun et al., 2025; Wang et al., 2025). Among these paradigms, diffusion-based generative frameworks have recently shown

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.

strong capability in modeling complex, high-dimensional data distributions, leading to notable progress in dance-to-music generation by improving synthesis quality and motion-music alignment. However, most existing methods condition music generation on motion using only coarse, global features, without explicitly modeling the rhythmic and temporal structure embedded in the dance movement. As a result, generated music often fails to capture fine-grained beat-level synchronization or onset-level transitions, leading to perceptible misalignment between the dance and the generated audio. Furthermore, the majority of prior works are constrained to fixed-length generation, severely limiting their practical applicability to real-world scenarios where dance performances vary widely in duration. Therefore, simultaneously satisfying the multi-scale alignment requirements while supporting arbitrary-length generation remains an open challenge.

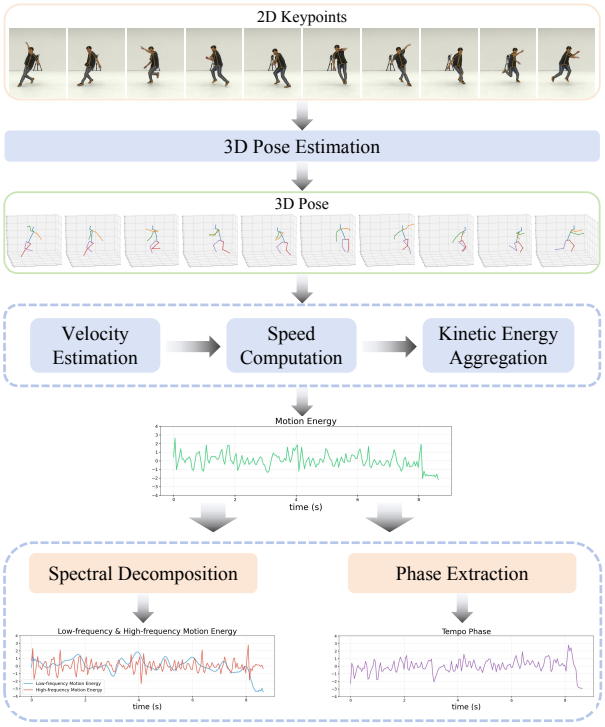


Figure 1: **Motion feature extraction.** Given a dance video, we estimate 3D body poses and compute the **per-frame kinetic energy**. The energy signal is decomposed using a Butterworth low-pass filter into a **low-frequency component**, capturing sustained body movements and global motion dynamics, and a **high-frequency component**, capturing rapid gestures and footwork. In addition, the **instantaneous rhythmic phase** is extracted from the energy signal using the Hilbert transform to represent the position within the beat cycle.

To address these limitations, we propose **PHE-DiT** (Phase-aware Hierarchical Energy Conditioning for Flexible-Length Dance-to-Music Generation), a conditional flow matching (Lipman et al., 2022) framework with a LLaMA-based (Touvron et al., 2023) cross-modal Transformer backbone for dance-to-music generation (as shown in Figs. 1 and 2). Our approach incorporates three key innovations that explicitly model rhythmic structure and motion dynamics to enable temporally coherent music synthesis from dance inputs of arbitrary length. **First**, we introduce *phase-aware input fusion*, which explicitly encodes the instantaneous phase ϕ extracted from the input dance motion via the Hilbert transform (Kschischang, 2006) as a pair of sinusoidal features $(\sin\phi, \cos\phi)$ and injects them into the input latent sequence. This design enables the model to perceive the current position within the beat cycle at every generation step, providing an explicit rhythmic clock signal derived directly from the dance movement. Crucially, our tempo phase encoding is complementary to Rotary Position Embedding (RoPE) (Heo et al., 2024). Specifically, whereas RoPE captures sequential positional order, our phase encoding explicitly represents rhythmic phase information. Together, these complementary mechanisms improve both beat-level synchronization and global tempo consistency between the generated music and the input motion. **Second**, we propose *hierarchical split energy conditioning*, which decomposes the dance motion energy into low-frequency (Slow Energy) and high-frequency (Fast Energy) components to provide multi-scale conditioning signals for music generation. Specifically, slow energy, obtained via Butterworth low-pass filtering (Hussin et al., 2016), captures global tempo and overall dynamic variations in the dancer movement, while fast energy, defined as the residual, encodes local transients and sharp motion onsets. Within the DiT architecture, the earlier transformer layers are conditioned on slow energy via cross-attention to establish coarse rhythmic structure in the generated music, while the later layers are conditioned on fast energy for fine-grained onset alignment. This coarse-to-fine conditioning strategy mirrors the multi-scale structure of music and simultaneously improves both beat-level and onset-level synchronization between the generated audio and the input dance. **Third**, we enable *flexible-length music generation* through variable-length sequence support. The model tracks effective sequence lengths for both audio and motion modalities,

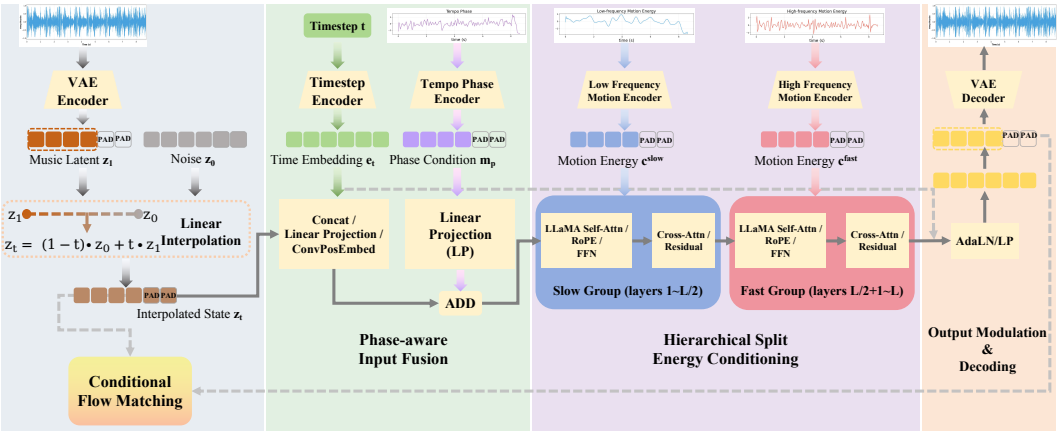


Figure 2: **Overview of PHE-DiT.** It comprises four parts: (1) *Conditional Flow Matching* defines the training objective by interpolating noise $\mathbf{z}_0$ and data $\mathbf{z}_1$ to learn a velocity field; (2) *Phase-aware Input Fusion* fuses the noisy latent, timestep embedding, and rhythmic phase encoding; (3) *Hierarchical Split Energy Conditioning* injects $\mathbf{c}^{\text{slow}}$ and $\mathbf{c}^{\text{fast}}$ into the lower and upper transformer layers via cross-attention, respectively; (4) *Output Modulation and Decoding* applies timestep-conditioned AdaLN and projects back to the audio latent space. PAD tokens enable flexible-length generation.

and applies length-aware attention masks to exclude padding tokens from both self-attention and cross-attention computations. At inference time, our framework naturally supports flexible-length generation conditioned on the duration of the input dance sequence. Specifically, the number of latent frames is determined dynamically based on the motion length, and the learned velocity field is integrated using an Euler ODE solver (Lipman et al., 2022) with classifier-free guidance (Ho & Salimans, 2022) to synthesize the final music representation. In summary, the main contributions of this paper are as follows:

- We propose phase-aware input fusion, which encodes instantaneous phase extracted from dance motion via the Hilbert transform as sinusoidal features and integrates them with the input latent representation via residual addition, providing an explicit rhythmic beat-cycle signal that is complementary to RoPE-based positional encoding.
- We introduce hierarchical split energy conditioning, which decomposes dance motion energy into slow and fast components and applies them hierarchically across transformer layers, achieving coarse-to-fine alignment between generated music and input motion at both beat and onset levels.
- We develop a flexible-length music generation framework via variable-length sequence support with length-aware attention masking, enabling a single model to generate music of any duration conditioned on dance sequences of varying length, without retraining or architectural modification.
- Extensive experiments demonstrate that PHE-DiT achieves state-of-the-art performance on standard benchmarks, generating music that exhibits superior rhythmic alignment with input dance motion, high audio realism, and flexible duration control.

2 Related Work

Dance-to-Music Generation. Dance-to-music (D2M) generation (Di et al., 2021; Zhu et al., 2022a;b; Luo et al., 2023; Yu et al., 2023; Sun et al., 2025; Wang et al., 2025) aims to synthesize music that is rhythmically synchronized and semantically consistent with dance movements. Unlike generic audio synthesis (Dhariwal et al., 2020; Liu et al., 2023; Agostinelli et al., 2023), D2M requires fine-grained synchronization with motion dynamics across multiple temporal scales, including global tempo consistency, beat-level correspondence, and onset-level precision. The inherent discrepancy between discrete motion representations and continuous audio waveforms further increases the difficulty of cross-modal synchronization, making D2M substantially more demanding than standard conditional music generation and motivating explicit modeling of rhythmic structure.

Non-autoregressive Dance-to-Music Generation Modeling. Early D2M works adopted adversarial or sequence modeling paradigms: D2M-GAN (Zhu et al., 2022a) learns cross-modal mappings through adversarial training with rhythm-aware objectives, while CMT (Di et al., 2021) employs

transformer-based sequence-to-sequence architectures to capture long-range motion-audio dependencies. More recently, diffusion-based frameworks (Zhu et al., 2022b; Luo et al., 2023; Yu et al., 2023; Sun et al., 2025; Wang et al., 2025) have become the dominant paradigm due to superior training stability and synthesis fidelity. CDCD (Zhu et al., 2022b) augments discrete diffusion with contrastive objectives to strengthen cross-modal alignment. LORIS (Yu et al., 2023) applies latent diffusion in compressed audio representations for structured and controllable synthesis. Diff-Foley (Luo et al., 2023) proposes a contrastive audio-visual pre-trained framework with double-guidance for temporally synchronized audio generation from visual dynamics. PN-Diffusion (Sun et al., 2025) introduces negative conditioning to suppress rhythmically inconsistent patterns during denoising. Collectively, these approaches have significantly improved generation fidelity and cross-modal temporal coherence.

Fixed-Length Dance-to-Music Generation. Existing D2M approaches (Zhu et al., 2022a;b; Sun et al., 2025) generally adopt a fixed-length training paradigm, where dance-music pairs are pre-segmented into short synchronized clips during both training and inference. For example, D2M-GAN (Zhu et al., 2022a) operates on 2-second clips, PN-Diffusion (Sun et al., 2025) on 5-second segments, and LORIS (Yu et al., 2023) on 25-second and 50-second clips. While this design simplifies optimization and batching, it confines generation to predefined temporal windows. In practice, dance videos often have variable durations, and the inability to generalize beyond fixed lengths restricts scalability to real-world applications such as long-form choreography. Although attention-based architectures can in principle handle variable-length inputs, most existing methods lack explicit mechanisms to maintain structural coherence over extended temporal horizons. In our work, we move beyond the fixed-length paradigm and formulate D2M generation as a flexible-length conditional sequence modeling problem.

3 METHOD

3.1 PROBLEM FORMULATION AND MOTION FEATURE EXTRACTION

Given a dance video $\mathcal{D}$ consisting of T frames, our goal is to generate a music audio signal that is rhythmically aligned with the input dance. Although 2D keypoints can represent skeletal trajectories in the image plane, they inherently lack depth information and cannot fully characterize forward-backward displacements or subtle body rotations. We therefore adopt a two-stage 3D pose extraction pipeline. First, we extract frame-level 2D skeletons using AlphaPose (Fang et al., 2022), obtaining $\mathbf{J}^{\text{2D}} \in \mathbb{R}^{T\times 24\times 3}$, where 24 denotes the number of body keypoints and each keypoint is represented by its spatial position together with an associated confidence score. We then reconstruct 3D skeletal joints via MotionBERT (Zhu et al., 2023), yielding $\mathbf{J}^{\text{3D}} \in \mathbb{R}^{T\times 17\times 3}$, where 17 is the number of reconstructed 3D keypoints and 3 corresponds to the spatial coordinates (x, y, z). The music generation is performed in the latent space of a pre-trained Variational Autoencoder (VAE) (Ning et al., 2025), which compresses the raw audio waveform into a compact latent representation $\mathbf{z} \in \mathbb{R}^{N\times d_z}$, where N is the latent sequence length and d_z is the latent channel dimension.

Motion Feature Extraction. We extract a compact motion feature representation from the 3D body poses through the following pipeline. First, we compute the per-joint velocity as the finite difference between consecutive frames: $\mathbf{v}_t = \mathbf{J}^{\text{3D}}_{t+1} - \mathbf{J}^{\text{3D}}_{t}$, and obtain the per-joint speed as the Euclidean norm $s_{t,j} = \|\mathbf{v}_{t,j}\|$, where $j \in \{0, \ldots, 16\}$ is the index of keypoint. The total body kinetic energy at each frame is then computed by summing squared speeds across all joints:

$$E_t = \sum_{j=0}^{16} s_{t,j}^2. \tag{1}$$

Spectral Decomposition. As shown in Figure 1, we decompose the energy signal into low-frequency and high-frequency components using a 4th-order Butterworth low-pass filter (Hussin et al., 2016) with a cutoff frequency of 2 Hz:

$$E^{\text{slow}} = \text{LPF}_{\text{Butter}}(E), \quad E^{\text{fast}} = E - E^{\text{slow}}, \tag{2}$$

where E^{slow} captures large, sustained movements (*e.g.*, body swaying), while E^{fast} captures rapid, fine-grained motions (*e.g.*, hand gestures, footwork). Both components are z-score normalized per clip and resampled from the video frame rate to the audio latent frame rate, yielding $\mathbf{m_s}, \mathbf{m_f} \in \mathbb{R}^{N\times 1}$.

Rhythmic Phase Extraction. We apply the Hilbert transform (Kschischang, 2006) to the body energy signal to obtain the instantaneous phase $\phi_t = \arg(\mathcal{H}(E_t))$, which encodes the rhythmic periodicity of the dance movement. The phase signal is likewise z-score normalized and resampled to the audio latent frame rate, yielding $\mathbf{m_p} \in \mathbb{R}^{N\times 1}$.

3.2 Conditional Flow Matching

We adopt conditional flow matching (CFM) (Tamir et al., 2024) as the generative framework, which defines a continuous-time probability path from noise to data in the learned latent space.

Training Objective. Let $\mathbf{z}_1 \in \mathbb{R}^{N\times d_z}$ denote a clean audio latent encoded by the pre-trained VAE and $\mathbf{z}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ a Gaussian noise sample. We construct the interpolated state at time $t \in [0, 1]$:

$$\mathbf{z}_t = (1-t)\mathbf{z}_0 + t\mathbf{z}_1. \tag{3}$$

The target velocity field is $\mathbf{u} = \mathbf{z}_1 - \mathbf{z}_0$. The model v_θ is trained to predict this velocity:

$$\mathcal{L} = \mathbb{E}_{t,\mathbf{z}_0,\mathbf{z}_1}\left[\left\|v_\theta(\mathbf{z}_t, \mathbf{m}, t) - \mathbf{u}\right\|^2\right], \tag{4}$$

where t is sampled via the logit-normal distribution $t = \sigma(\mathcal{N}(0, 1))$, and $\mathbf{m} = \{\mathbf{m_s}, \mathbf{m_f}, \mathbf{m_p}\}$ denotes the motion condition. Since the model supports variable-length sequences within each mini-batch (detailed in Section 3.5), shorter samples are right-padded to the maximum batch length. To prevent gradient signals from these uninformative padding positions, the loss is computed exclusively over valid (non-padding) temporal positions.

Classifier-Free Guidance. During training, the motion condition is randomly dropped with probability p_{drop}, replacing $\mathbf{m_s}$ and $\mathbf{m_f}$ with zeros. At inference, we apply classifier-free guidance (CFG):

$$\hat{v} = v_\theta(\mathbf{z}_t, \mathbf{m}, t) + s \cdot \left(v_\theta(\mathbf{z}_t, \mathbf{m}, t) - v_\theta(\mathbf{z}_t, \mathbf{0}, t)\right), \tag{5}$$

where s is the guidance scale that controls the strength of motion conditioning.

3.3 Phase-aware Input Fusion

Standard positional encodings such as Rotary Position Embedding (RoPE) encode the absolute or relative *order* of tokens within a sequence, enabling the model to distinguish temporal positions. However, they do not capture the *cyclic rhythmic structure* inherent in music, where events recur periodically within each beat cycle. As a result, relying solely on positional encodings makes it difficult for the model to perceive its current position within the beat period, leading to gradual drift in tempo consistency over long horizons.

To address this, we introduce phase-aware input fusion, which explicitly injects tempo phase information into the input sequence representation. Specifically, given the phase condition $\mathbf{m_p} \in \mathbb{R}^{N\times 1}$ extracted in Section 3.1, we encode it at each temporal position l as a pair of sinusoidal features:

$$\mathbf{p}_l = [\sin(\mathbf{m_p}_l),\ \cos(\mathbf{m_p}_l)] \in \mathbb{R}^2. \tag{6}$$

This circular encoding preserves the periodic continuity of the beat cycle, ensuring that phases near 0 and 2π are represented as neighboring points in the embedding space. The phase features are then added to the input token embeddings as a residual. Specifically, let $\mathbf{x} \in \mathbb{R}^{N\times d_x}$ denote the input sequence obtained by the input embedding layer, which concatenates the noisy audio latent $\mathbf{z}_t \in \mathbb{R}^{N\times d_z}$ and the broadcasted timestep embedding $\mathbf{e}_t \in \mathbb{R}^{d_t}$ along the channel dimension at each temporal position, projects the concatenated representation to the transformer's hidden dimension d via a linear layer, and adds a convolutional position embedding:

$$\mathbf{x} = \text{ConvPosEmbed}(W_{\text{in}} \cdot [\mathbf{z}_t \,\|\, \mathbf{e}_t]) + W_{\text{in}} \cdot [\mathbf{z}_t \,\|\, \mathbf{e}_t], \tag{7}$$

where $[\cdot \,\|\, \cdot]$ denotes concatenation along the channel dimension and $W_{\text{in}} \in \mathbb{R}^{(d_z+d_t)\times d}$ is a learnable projection. The phase features $\mathbf{P} = [\mathbf{p}_0, \ldots, \mathbf{p}_{N-1}]^\top \in \mathbb{R}^{N\times 2}$ are then projected via a learnable linear layer $W_{\text{phase}} \in \mathbb{R}^{2\times d}$ and added to $\mathbf{x}$ as a residual:

$$\tilde{\mathbf{x}} = \mathbf{x} + W_{\text{phase}} \cdot \mathbf{P}. \tag{8}$$

The fused representation $\tilde{\mathbf{x}} \in \mathbb{R}^{N\times d}$ is then fed into the transformer blocks, where RoPE is subsequently applied to the query and key projections within each self-attention layer. Crucially, our

phase encoding is *complementary* to RoPE: while RoPE encodes sequence order (where am I in the sequence?), phase encoding captures rhythmic position (where am I in the current beat cycle?). Together, they provide the model with both sequential awareness and rhythmic awareness, improving beat-level alignment and global tempo consistency between the generated music and the input motion.

3.4 Hierarchical Split Energy Conditioning

Music exhibits a natural multi-scale temporal hierarchy, where global tempo defines the overall rhythmic pace, beats provide intermediate periodic accents, and onsets capture fine-grained transient events. To mirror this structure in our model, we propose hierarchical split energy conditioning, which injects the decomposed energy components into different layers of the transformer via cross-attention.

Motion Energy Embedding. Each energy component is independently projected into the transformer's hidden space via a two-layer MLP with SiLU activation:

$$\mathbf{c}^{\text{slow}} = \text{MLP}_{\text{slow}}(\mathbf{m_s}), \quad \mathbf{c}^{\text{fast}} = \text{MLP}_{\text{fast}}(\mathbf{m_f}), \tag{9}$$

where $\mathbf{c}^{\text{slow}}, \mathbf{c}^{\text{fast}} \in \mathbb{R}^{N \times d_c}$ are the projected energy embeddings that serve as keys and values for cross-attention.

Hierarchical Cross-Attention Conditioning. We inject the two energy components into different layers of the DiT backbone to establish a coarse-to-fine conditioning pathway. Let the transformer consist of L cross-modal decoder layers indexed by $n \in \{1, \ldots, L\}$. We designate the first $L/2$ layers as the *slow group* and the remaining $L/2$ layers as the *fast group*. Each cross-modal decoder layer consists of a LLaMA self-attention block with RoPE followed by a cross-attention module:

$$\mathbf{h}^{(n)} = \text{LLaMA-SelfAttn}(\mathbf{h}^{(n-1)}), \tag{10}$$

$$\mathbf{h}^{(n)} = \mathbf{h}^{(n)} + \text{CrossAttn}(\text{LN}(\mathbf{h}^{(n)}),\ \mathbf{c}^{(n)}), \tag{11}$$

where $\text{LN}(\cdot)$ denotes layer normalization, and the cross-attention context $\mathbf{c}^{(n)}$ is selected based on the layer index:

$$\mathbf{c}^{(n)} = \begin{cases} \mathbf{c}^{\text{slow}}, & \text{if } n \leq L/2, \\ \mathbf{c}^{\text{fast}}, & \text{if } n > L/2. \end{cases} \tag{12}$$

In the cross-attention module, queries are derived from the audio hidden states, while keys and values are derived from the corresponding energy embedding, enabling the model to attend to motion dynamics at the appropriate temporal scale. This hierarchical design encourages the lower layers to capture coarse, structural correspondences between motion and music (*e.g.*, segment-level dynamics and tempo envelope), while the upper layers model fine-grained rhythmic alignment (*e.g.*, beat-level synchronization and onset precision). This progressive conditioning strategy explicitly mirrors the multi-scale hierarchy of music (tempo $\rightarrow$ beat $\rightarrow$ onset) and simultaneously improves both beat-level and onset-level synchronization.

Output Modulation. The final hidden states are modulated by an Adaptive Layer Normalization (AdaLN) conditioned on the timestep embedding, then projected back to the audio latent dimension:

$$\hat{\mathbf{u}} = W_{\text{out}} \cdot \text{AdaLN}(\mathbf{h}^{(L)}, \mathbf{e}_t), \tag{13}$$

where $\hat{\mathbf{u}} \in \mathbb{R}^{N \times d_z}$ is the predicted velocity field in the audio latent space, $\mathbf{e}_t$ is the sinusoidal timestep embedding, and $W_{\text{out}} \in \mathbb{R}^{d \times d_z}$ is a learnable output projection.

3.5 Flexible-Length Generation

Existing D2M methods typically operate under fixed-length training protocols, where dance-music pairs are pre-segmented into clips of predetermined duration. This restricts their applicability to real-world scenarios where dance performances vary widely in duration. To overcome this limitation, we design PHE-DiT to support variable-length sequence processing within a single unified model.

Length-aware Sequence Handling. During training, we construct mini-batches containing dance-music pairs of varying lengths. Let $N_{\max}$ denote the maximum latent sequence length within a mini-batch. Each sample is associated with dedicated length indicators l_{audio} and l_{motion} that specify the effective (non-padded) sequence lengths for the audio and motion modalities, respectively.

For each sample, both the audio latent $\mathbf{z}_1 \in \mathbb{R}^{l_{\text{audio}} \times d_z}$ and the motion conditions $\mathbf{m_s}, \mathbf{m_f}, \mathbf{m_p} \in \mathbb{R}^{l_{\text{motion}} \times 1}$ are right-padded with zeros to match $N_{\max}$, yielding uniform-length tensors within the batch. Correspondingly, the sampled noise $\mathbf{z}_0$ is also zeroed out at padding positions to ensure the interpolated state $\mathbf{z}_t$ (Eq. (3)) remains zero beyond the valid length, preventing the model from learning spurious patterns from padded regions.

Attention Masking. To prevent padded tokens from influencing the generation process, we apply length-aware attention masks to both self-attention and cross-attention computations. Specifically, for a sample with effective audio length l_{audio} and motion length l_{motion}, the attention masks are defined as:

$$\mathbf{M}_{\text{self}}[i,j] = \begin{cases} 0, & \text{if } i \le l_{\text{audio}} \text{ and } j \le l_{\text{audio}}, \\ -\infty, & \text{otherwise}, \end{cases} \tag{14}$$

$$\mathbf{M}_{\text{cross}}[i,j] = \begin{cases} 0, & \text{if } i \le l_{\text{audio}} \text{ and } j \le l_{\text{motion}}, \\ -\infty, & \text{otherwise}, \end{cases} \tag{15}$$

where $-\infty$ masks are applied before the softmax operation to exclude padded positions from attention computation. The training loss (Eq. (4)) is likewise computed only on valid (non-padding) positions, ensuring that the model learns exclusively from meaningful audio-motion correspondences.

Inference. At inference time, the number of generated latent frames is determined dynamically from the length of the input dance sequence, without any fixed-length constraint. Given an input motion sequence of arbitrary length, we sample initial noise $\mathbf{z}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and integrate the learned velocity field using the Euler ODE solver over S steps:

$$\mathbf{z}_{k+1} = \mathbf{z}_k + \frac{1}{S}\hat{v}(\mathbf{z}_k, \mathbf{m}, t_k), \tag{16}$$

where $\hat{v}$ is the classifier-free guided velocity (Eq. (5)). The final latent $\mathbf{z}_S$ is cropped to the valid length and decoded to a waveform by the pre-trained VAE decoder. This design enables seamless music generation from short dance clips to full-length choreographies within a single model, without requiring retraining or architectural modification.

4 Experiment

4.1 Experiment Setting

Datasets We evaluate on two benchmarks following the partitions and protocols of Sun et al. (2025). **AIST++** (Tsuchida et al., 2019) contains 1,020 studio-recorded videos across 10 dance genres, split into 8,819 training clips (7-48 s, avg. 14.25 s) and 180 test clips (7-12 s, avg. 9.38 s). **TikTok** (Zhu et al., 2022a) is curated from in-the-wild short videos with 85 unique songs, split into 391 training (3-28 s) and 53 test clips (5-26 s), providing a complementary resource for evaluating generalization.

Baselines. We compare PHE-DiT with six recent state-of-the-art approaches: Diff-Foley (Luo et al., 2023), D2M-GAN (Zhu et al., 2022a), CMT (Di et al., 2021), CDCD (Zhu et al., 2022b), LORIS (Yu et al., 2023), and PN-Diffusion (Sun et al., 2025). All six baselines are evaluated on AIST++, while D2M-GAN, CMT, CDCD, and PN-Diffusion are additionally evaluated on TikTok following the protocol of (Sun et al., 2025). These methods are selected because they are highly relevant to our D2M problem setting and provide publicly available implementations, pretrained weights, and hyper-parameters, which ensures reproducibility and fair comparison.

Evaluation Protocols. Following Zhu et al. (2022b); Sun et al. (2025), we evaluate rhythmic alignment at two levels: (1) **beat-level** for global tempo consistency and (2) **onset-level** for fine-grained transient synchronization, using 50 ms tolerance with greedy one-to-one matching. We report **BCS** (precision), **BHS** (recall), **F1**, and their standard deviations **CSD/HSD** for stability. We also adopt **Fréchet Audio Distance (FAD)** (Kilgour et al., 2018) with four extractors: VGGish (Diwakar & Gupta, 2024) (FAD_v), PANNs (Kong et al., 2020a) (FAD_p), CLAP (Elizalde et al., 2023) (FAD_c), and EnCodec (Défossez et al., 2022) (FAD_e). Additionally, subjective evaluation results are provided in Section A, and generated music examples are provided in the **supplementary material**.

Implementation Details. The architecture, training configuration, and sampling settings are provided in Appendix B.

Table 1: Rhythmic alignment on the AIST++ and TikTok test sets, evaluated at the beat and onset levels. The best performance for each metric is highlighted in bold.

Dataset	Method	Beat-level					Onset-level				
		BCS↑	CSD↓	BHS↑	HSD↓	F1↑	BCS↑	CSD↓	BHS↑	HSD↓	F1↑
AIST++	Diff-foley (Luo et al., 2023)	19.03	16.31	20.18	16.41	18.52	20.12	16.62	25.41	19.63	20.28
	D2M-GAN (Zhu et al., 2022a)	18.75	21.55	21.40	24.18	18.92	38.08	23.61	31.77	23.64	30.53
	CMT (Di et al., 2021)	31.87	24.27	31.99	26.17	29.91	41.87	33.20	24.68	20.58	25.49
	CDCD (Zhu et al., 2022b)	25.51	19.71	27.39	21.77	26.23	45.15	15.10	43.17	20.32	39.63
	LORIS (Yu et al., 2023)	34.78	**7.05**	48.07	**11.01**	39.58	24.43	20.42	43.85	20.84	25.06
	PN-Diffusion (Sun et al., 2025)	39.70	17.86	45.66	18.34	41.02	42.65	21.87	47.76	24.69	44.00
	PHE-DiT (Ours)	**70.50**	15.28	**80.33**	15.65	**74.00**	**48.80**	**12.45**	**51.01**	**14.47**	**50.44**
TikTok	D2M-GAN (Zhu et al., 2022a)	45.12	28.16	37.39	25.32	36.02	34.71	27.40	41.16	32.79	35.22
	CMT (Di et al., 2021)	40.11	25.12	43.44	**10.06**	45.45	40.11	25.12	43.44	14.78	39.97
	CDCD (Zhu et al., 2022b)	59.68	25.88	52.42	26.50	51.37	48.54	34.84	55.07	38.06	49.20
	PN-Diffusion (Sun et al., 2025)	52.31	**22.13**	45.51	14.96	50.99	67.30	21.60	72.38	20.14	70.69
	PHE-DiT (Ours)	**67.73**	25.13	**55.04**	10.88	**51.64**	**78.28**	**21.09**	**87.83**	**13.88**	**79.91**

4.2 Main Results

Rhythmic Alignment. Table 1 presents the quantitative comparison on both AIST++ and TikTok test sets at beat-level and onset-level alignment. First, on AIST++, PHE-DiT achieves a beat-level F1 of 74.00, surpassing the strongest baseline PN-Diffusion (41.02) by a large margin of $+32.98$ points, with BCS and BHS reaching 70.50 and 80.33, respectively. This substantial improvement in beat-level alignment validates the effectiveness of our *phase-aware input fusion*, which provides an explicit rhythmic clock signal via the Hilbert-transform-derived phase encoding, enabling the model to maintain precise global tempo consistency. At the onset-level, PHE-DiT also achieves the best F1 of 50.44 (vs. 44.00 for PN-Diffusion), with the lowest CSD (12.45) and HSD (14.47) among all methods, indicating superior generation stability. This confirms that our *hierarchical split energy conditioning* effectively captures fine-grained transient alignment by routing high-frequency motion energy to the upper transformer layers. Second, on TikTok, PHE-DiT achieves onset-level F1 79.91, outperforming PN-Diffusion (70.69) by $+9.22$ points, while attaining the best BCS (78.28), BHS (87.83), CSD (21.09), and HSD (13.88). At beat-level, PHE-DiT achieves the highest BCS (67.73) and F1 (51.64). The consistent improvements on TikTok, which contains in-the-wild dance videos with diverse styles and variable durations, demonstrate the strong generalization of our *flexible-length generation* framework.

Audio Quality. Table 2 reports the Fréchet Audio Distance (FAD) evaluated with four embedding extractors. On AIST++, PHE-DiT achieves the best FAD across all four extractors: FAD_v=3.89 (vs. 6.31 for PN-Diffusion), FAD_p=0.22, FAD_c=0.58, and FAD_e=93.42, demonstrating that the generated music closely matches the distribution of real music. On TikTok, PHE-DiT similarly achieves the lowest FAD_v (11.51), FAD_c (0.96), and FAD_e (51.16). These results indicate that our model not only improves rhythmic alignment but also generates audio of higher perceptual quality, benefiting from the motion-conditioned generation framework that produces musically coherent outputs. Moreover, additional **music examples** comparing music generated by our method and baseline approaches are available in the **supplementary material**.

Table 2: Comparison of generated music quality using Fréchet Audio Distance (FAD). FAD evaluates whether generated music matches the distribution of real music in a learned embedding space, without requiring human annotations. We compare PHE-DiT with baselines on the AIST++ and TikTok test sets. Lower FAD is better, and the best scores are highlighted in bold.

Dataset	Method	FAD_v↓	FAD_p ($\times 10^{-4}$)↓	FAD_c↓	FAD_e↓
AIST++	Diff-foley (Luo et al., 2023)	27.77	21.75	1.52	535.86
	D2M-GAN (Zhu et al., 2022a)	11.29	0.30	1.49	666.09
	CMT (Di et al., 2021)	12.57	0.25	1.02	584.43
	CDCD (Zhu et al., 2022b)	7.48	1.81	1.26	255.12
	LORIS (Yu et al., 2023)	7.71	3.98	1.03	457.34
	PN-Diffusion (Sun et al., 2025)	6.31	0.30	0.65	103.56
	PHE-DiT (Ours)	**3.89**	**0.22**	**0.58**	**93.42**
TikTok	D2M-GAN (Zhu et al., 2022a)	27.30	1.33	1.46	254.83
	CMT (Di et al., 2021)	25.00	1.35	1.34	287.45
	CDCD (Zhu et al., 2022b)	14.99	**0.50**	1.25	111.46
	PN-Diffusion (Sun et al., 2025)	12.37	11.43	1.19	60.53
	PHE-DiT (Ours)	**11.51**	2.26	**0.96**	**51.16**

4.3 Ablation Studies

To validate the contribution of each proposed component, we conduct systematic ablation studies on the AIST++ dataset. Results are reported in Table 3.

Effect of Phase-aware Input Fusion. Removing the phase encoding ("w/o Phase") increases beat-level CSD from 15.28 to 26.85 ($+11.57$) and onset-level CSD from 12.45 to 27.03 ($+14.58$), revealing a substantial degradation in generation stability at both temporal scales. Onset-level F1

Table 3: Ablation study on the AIST++ test set. We evaluate the contribution of each component by removing or modifying modules from the full PHE-DiT model. Best results are highlighted in **bold**.

Variant	Beat-level					Onset-level				
	BCS↑	CSD↓	BHS↑	HSD↓	F1↑	BCS↑	CSD↓	BHS↑	HSD↓	F1↑
PHE-DiT (Full)	**70.50**	15.28	**80.33**	15.65	**74.00**	**48.80**	**12.45**	**51.01**	**14.47**	**50.44**
w/o Phase	66.22	26.85	76.38	26.12	69.87	35.97	27.03	47.61	26.60	34.56
w/o Hierarchical split	61.71	32.50	70.18	35.14	64.55	45.68	29.50	55.12	21.99	43.48
Reversed split order	64.81	34.93	71.86	36.69	67.32	40.20	29.64	42.61	21.55	41.09
Slow energy only	64.72	23.48	73.92	25.48	67.96	43.31	30.80	50.07	15.49	40.55
Fast energy only	67.21	29.44	76.07	21.00	70.29	43.89	17.78	45.59	25.35	41.54
w/o Self-attn	17.00	8.02	21.62	8.99	18.56	23.49	12.76	32.02	11.21	25.20
w/o Self-attn + Phase	16.32	**7.94**	20.43	**8.96**	17.74	23.32	12.74	32.13	10.32	25.18

drops sharply from 50.44 to 34.56 (-15.88), with BCS decreasing from 48.80 to 35.97. These results demonstrate that the Hilbert-transform-derived phase signal serves as an effective rhythmic clock that stabilizes generation across beat cycles. Notably, while RoPE provides sequential position awareness, it cannot capture the *cyclic periodicity* of beat structure, and the phase encoding complements RoPE by explicitly informing the model of its position within each beat cycle, leading to more consistent rhythmic patterns.

Effect of Hierarchical Split Energy Conditioning. We evaluate four variants to assess the hierarchical conditioning design. (1) "w/o Hierarchical split" applies both energy components uniformly to all layers. While onset-level BHS (55.12) is higher than the full model (51.01), the CSD increases dramatically from 12.45 to 29.50 and HSD from 14.47 to 21.99, indicating highly unstable generation. Onset-level F1 drops from 50.44 to 43.48 (-6.96), and beat-level F1 also drops by 9.45 points ($74.00 \rightarrow 64.55$). This reveals that without hierarchical separation, the model may occasionally produce high-recall outputs but lacks the consistency needed for reliable generation. (2) "Reversed split order" swaps the slow/fast layer assignment, yielding beat-level CSD of 34.93 and HSD of 36.69, which is the worst stability among all energy-conditioning variants, confirming that the specific assignment of slow energy to lower layers and fast energy to upper layers is not arbitrary but reflects the natural coarse-to-fine processing hierarchy of the transformer. (3) "Slow energy only" and (4) "Fast energy only" each condition all layers with a single energy component. Both exhibit substantially worse CSD/HSD compared to the full model at both levels (*e.g.*, beat CSD: 23.48 and 29.44 vs. 15.28), validating that both energy components contribute complementary information and that the hierarchical split is necessary for optimal alignment.

Effect of Self-Attention. Removing self-attention ("w/o Self-attn") causes catastrophic performance collapse, with beat-level F1 dropping to 18.56 and onset-level F1 to 25.20, regardless of whether phase encoding is included (17.74 and 25.18 with phase). This demonstrates that self-attention is the fundamental mechanism for modeling temporal dependencies in the audio latent sequence, and that neither cross-attention conditioning nor phase encoding can compensate for its absence.

Effect of Layer Split Ratio. The balanced split ($K = 8$ of 16 layers) achieves the best performance among the tested allocations. The complete results and analysis are provided in Appendix C.

4.4 Qualitative Analysis

Figure A1 in Appendix D compares rhythmic alignment on AIST++ test samples. PHE-DiT produces denser and better-aligned beat and onset events over the full input sequence ($\sim$11 s), while the baselines operate on fixed-length 5 s segments. These examples complement the quantitative results and illustrate flexible-length generation.

5 Conclusion

We present PHE-DiT, a conditional flow matching framework for dance-to-music generation that explicitly models rhythmic structure and supports flexible-length synthesis. It integrates three components: (1) phase-aware input fusion that encodes Hilbert-transform-derived rhythmic phase as a beat-cycle clock signal complementary to positional encodings; (2) hierarchical split energy conditioning that routes slow and fast motion energy to lower and upper transformer layers for coarse-to-fine alignment; and (3) length-aware attention masking that enables arbitrary-duration generation without retraining. Experiments on AIST++ and TikTok show that PHE-DiT achieves state-of-the-art rhythmic alignment and audio quality.

AI USE STATEMENT

In this work, generative AI tools were used exclusively to polish the language of this manuscript, correct grammatical and spelling errors, and improve clarity and readability. Research conceptualization, method development, and data analysis were conducted without AI tools. All language revisions were reviewed by the authors, who retain full responsibility for the content of the manuscript.

REFERENCES

Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, et al. Musiclm: Generating music from text. *arXiv preprint arXiv:2301.11325*, 2023.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation. *Advances in neural information processing systems*, 36:47704–47720, 2023.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*, 2022.

Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

Shangzhe Di, Zeren Jiang, Si Liu, Zhaokai Wang, Leyan Zhu, Zexin He, Hongming Liu, and Shuicheng Yan. Video background music generation with controllable music transformer. In *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 2037–2045, 2021.

Mandar Pramod Diwakar and Brijendra Gupta. Vggish deep learning model: Audio feature extraction and analysis. In *International Conference on Data Management, Analytics & Innovation*, pp. 59–70. Springer, 2024.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. Clap learning audio concepts from natural language supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023.

Hao-Shu Fang, Jiefeng Li, Hongyang Tang, Chao Xu, Haoyi Zhu, Yuliang Xiu, Yong-Lu Li, and Cewu Lu. Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time. *IEEE transactions on pattern analysis and machine intelligence*, 45(6):7157–7173, 2022.

Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer. In *European Conference on Computer Vision*, pp. 289–305. Springer, 2024.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Siti Farah Hussin, Gauri Birasamy, and Zunainah Hamid. Design of butterworth band-pass filter. *Politeknik & Kolej Komuniti Journal of Engineering and Technology*, 1(1):32–46, 2016.

Jaekwon Im, Natalia Polouliakh, and Taketo Akama. Pf-d2m: A pose-free diffusion model for universal dance-to-music generation. *arXiv preprint arXiv:2601.15872*, 2026.

Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. *Advances in Neural Information Processing Systems*, 36:20067–20079, 2023.

Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. Fr\'echet audio distance: A metric for evaluating music enhancement algorithms. *arXiv preprint arXiv:1812.08466*, 2018.

Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley. Panns: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894, 2020a.

Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761*, 2020b.

Frank R Kschischang. The hilbert transform. *University of Toronto*, 83:277, 2006.

Max WY Lam, Qiao Tian, Tang Li, Zongyu Yin, Siyuan Feng, Ming Tu, Yuliang Ji, Rui Xia, Mingbo Ma, Xuchen Song, et al. Efficient neural music generation. *Advances in Neural Information Processing Systems*, 36:17450–17463, 2023.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. *arXiv preprint arXiv:2301.12503*, 2023.

Simian Luo, Chuanhao Yan, Chenxu Hu, and Hang Zhao. Diff-foley: Synchronized video-to-audio synthesis with latent diffusion models. *Advances in Neural Information Processing Systems*, 36: 48855–48876, 2023.

Ziqian Ning, Huakang Chen, Yuepeng Jiang, Chunbo Hao, Guobin Ma, Shuai Wang, Jixun Yao, and Lei Xie. Diffrhythm: Blazingly fast and embarrassingly simple end-to-end full-length song generation with latent diffusion. *arXiv preprint arXiv:2503.01183*, 2025.

Robert C Streijl, Stefan Winkler, and David S Hands. Mean opinion score (mos) revisited: methods and applications, limitations and alternatives. *Multimedia Systems*, 22(2):213–227, 2016.

Changchang Sun, Gaowen Liu, Charles Fleming, and Yan Yan. Enhancing dance-to-music generation via negative conditioning latent diffusion model. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8321–8330, 2025.

Ella Tamir, Najwa Laabid, Markus Heinonen, Vikas Garg, and Arno Solin. Conditional flow matching for time series modelling. In *ICML 2024 Workshop on Structured Probabilistic Inference {\&} Generative Modeling*, 2024.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Shuhei Tsuchida, Satoru Fukayama, Masahiro Hamasaki, and Masataka Goto. Aist dance video database: Multi-genre, multi-dancer, and multi-camera database for dance information processing. In *ISMIR*, volume 1, pp. 6, 2019.

Aaron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu, et al. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 12(1), 2016.

Jinting Wang, Chenxing Li, and Li Liu. Gaca-dit: Diffusion-based dance-to-music generation with genre-adaptive rhythm and context-aware alignment. *arXiv preprint arXiv:2510.26818*, 2025.

Jiashuo Yu, Yaohui Wang, Xinyuan Chen, Xiao Sun, and Yu Qiao. Long-term rhythmic video soundtracker. In *International Conference on Machine Learning*, pp. 40339–40353. PMLR, 2023.

Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. Motionbert: A unified perspective on learning human motion representations. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 15085–15099, 2023.

Ye Zhu, Kyle Olszewski, Yu Wu, Panos Achlioptas, Menglei Chai, Yan Yan, and Sergey Tulyakov. Quantized gan for complex music generation from dance videos. In *European Conference on Computer Vision*, pp. 182–199. Springer, 2022a.

Ye Zhu, Yu Wu, Kyle Olszewski, Jian Ren, Sergey Tulyakov, and Yan Yan. Discrete contrastive diffusion for cross-modal music and image generation. *arXiv preprint arXiv:2206.07771*, 2022b.

APPENDIX

A SUBJECTIVE EVALUATION

Following Sun et al. (2025), we conduct a user study on 30 randomly selected AIST++ test samples. 10 volunteers are asked to watch each dance video paired with the generated music and rate two aspects on a 1–5 Likert scale: **Overall Quality (OVL)**, assessing the naturalness and fidelity of the generated audio, and **Relevance (REL)**, assessing the perceived rhythmic correspondence between the music and the dance movements. We report the Mean Opinion Score (MOS) (Streijl et al., 2016) for each dimension. Additionally, we perform a Turing Test in which volunteers are presented with both generated and real music samples and asked to identify which is real; the proportion of generated samples judged as real serves as a measure of perceptual realism.

As shown in Table A1, PHE-DiT achieves the highest Mean Opinion Scores (MOS) in both OVL and REL dimensions, indicating that our generated music not only sounds more realistic but also exhibits stronger perceived rhythmic correspondence with the input dance. In the Turing Test, PHE-DiT obtains the highest proportion of samples judged as real, further confirming the perceptual quality of our approach.

Table A1: Subjective evaluation results (MOS) on the AIST++ test set. OVL measures overall audio quality and REL measures rhythmic relevance to the dance. Higher is better.

Method	OVL↑	REL↑	Turing (%)↑
CMT (Di et al., 2021)	2.8	2.3	0.20
CDCD (Zhu et al., 2022b)	2.9	2.8	0.31
PN-Diffusion (Sun et al., 2025)	3.9	4.0	0.51
PHE-DiT (Ours)	4.4	4.3	0.67

B IMPLEMENTATION DETAILS

We use the pre-trained VAE from Ning et al. (2025) (d_z=64, frame rate ≈21.5 Hz, compression f=2048). The DiT backbone has L=16 cross-modal decoder layers (LLaMA self-attention with RoPE + cross-attention), hidden dimension d=2048, 32 heads. Energy embeddings are projected to d_c=512 via two-layer MLPs with SiLU. The first 8 layers are conditioned on slow energy, the remaining 8 on fast energy. The Butterworth filter is 4th-order with cutoff f_c=2 Hz. Training uses batch size 2 per GPU on 4 GPUs, logit-normal timestep sampling, and condition dropout p_{drop}=0.2. Inference uses the Euler ODE solver with S=32 steps and guidance scale s=4.0.

C EFFECT OF LAYER SPLIT RATIO

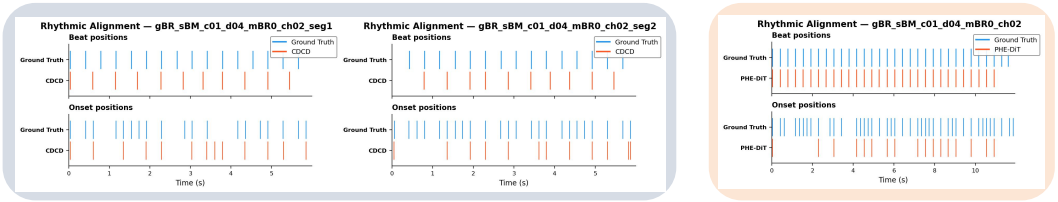
We also investigate the impact of the layer split ratio K that determines the boundary between slow-energy and fast-energy conditioned layers. With $L = 16$ total layers, we evaluate $K \in \{0, 4, 8, 12, 16\}$, where $K = 0$ and $K = 16$ correspond to using only fast or only slow energy for all layers, respectively. Results are shown in Table A2. The balanced split $K = 8$ achieves the best performance across all metrics, with beat-level F1 of 74.00 and onset-level F1 of 50.44. When too few layers are assigned to slow energy ($K = 0$ or $K = 4$), beat-level alignment degrades notably (F1: 70.29 and 66.77), as the model lacks sufficient capacity to establish global tempo structure. Conversely, when too many layers receive slow energy ($K = 12$ or $K = 16$), onset-level CSD increases sharply to 28.64 and 30.80 (vs. 12.45 for $K = 8$), indicating that reducing the fast-energy pathway impairs fine-grained transient alignment. This confirms that an equal split effectively balances coarse rhythmic modeling in the lower layers with precise onset tracking in the upper layers.

D QUALITATIVE ANALYSIS

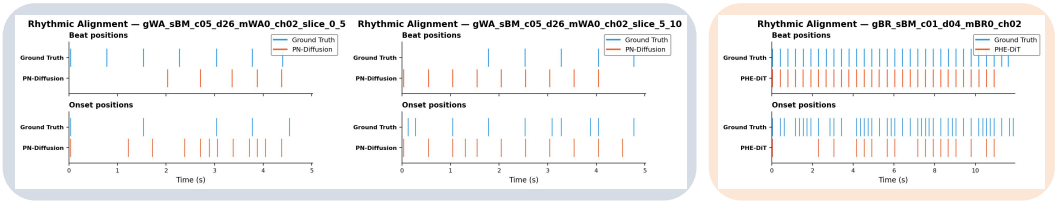
Visualization of Rhythmic Alignment. To provide an intuitive understanding of the alignment quality, we visualize representative examples from the AIST++ test set in Fig. A1. For each sample,

Table A2: Effect of layer split ratio K on AIST++. K determines how many of the 16 layers are conditioned on slow energy (layers 1 to K) vs. fast energy (layers $K+1$ to 16).

Split ratio K	Beat-level					Onset-level				
	BCS↑	CSD↓	BHS↑	HSD↓	F1↑	BCS↑	CSD↓	BHS↑	HSD↓	F1↑
$K = 0$ (0%/100%)	67.21	29.44	76.07	21.00	70.29	43.89	17.78	45.59	25.35	41.54
$K = 4$ (25%/75%)	63.34	21.63	73.08	23.80	66.77	46.41	19.82	50.54	19.87	45.14
$K = 8$ (50%/50%)	**70.50**	**15.28**	**80.33**	**15.65**	**74.00**	**48.80**	**12.45**	**51.01**	**14.47**	**50.44**
$K = 12$ (75%/25%)	68.33	19.86	78.77	20.92	72.61	47.31	28.64	50.40	17.02	41.29
$K = 16$ (100%/0%)	64.72	23.48	73.92	25.48	67.96	43.31	30.80	50.07	15.49	40.55



(a) PHE-DiT (Ours) vs. CDCD



(b) PHE-DiT (Ours) vs. PN-Diffusion

Figure A1: Qualitative comparison of rhythmic alignment on AIST++ test samples. Blue and red lines denote ground-truth and generated beat/onset positions, respectively. Baselines (left, blue panels) operate on fixed-length 5 s segments, while PHE-DiT (right, orange panel) generates over the full sequence. PHE-DiT produces denser and better-aligned beat/onset events.

we plot the ground-truth beat/onset positions (blue) alongside those detected from the generated music (red). As shown in Fig. A1a, CDCD generates beats and onsets that are noticeably sparse and temporally shifted relative to the ground truth, with many ground-truth events entirely missed across both 5 s segments. In Fig. A1b, PN-Diffusion exhibits even sparser detections, particularly at the onset level, where very few generated events correspond to ground-truth transients. In contrast, PHE-DiT (right panels) produces substantially denser beat and onset events that closely align with the ground-truth positions over the full sequence duration ($\sim$11 s), without requiring segmentation into fixed-length clips. These qualitative results corroborate the quantitative improvements reported in Table 1 and highlight two key advantages of our approach: the phase-aware input fusion enables PHE-DiT to maintain consistent beat alignment across the entire sequence, while the hierarchical split energy conditioning captures fine-grained onset events that baselines fail to detect. Moreover, the flexible-length generation capability allows PHE-DiT to operate directly on the full dance sequence, avoiding the boundary artifacts inherent in segment-based approaches.