A Data-Driven Method for Analyzing and Quantifying Lyrics-Dance Motion Relationships

Kento Watanabe and Masataka Goto
National Institute of Advanced Industrial Science and Technology (AIST)
{kento.watanabe, m.goto}@aist.go.jp

Abstract

Dancing to music with lyrics is a popular form of expression. While it is generally accepted that there are relationships between lyrics and dance motions, previous studies have not explored these relationships. A major challenge is that the relationships between lyrics and dance motions are not constant throughout a song but are instead localized to specific parts. To address this challenge, we hypothesize that lyrics and dance motions that co-occur across multiple songs are related. Based on this hypothesis, we propose a novel data-driven method to detect the parts of songs where meaningful relationships between lyrics and dance motions exist. We use clustering to transform lyrics and dance motions into symbols, enabling the calculation of co-occurrence frequencies and detection of significant correlations. The effectiveness of our method is validated by a dataset of time-synchronized lyrics and dance motions, which showed high correlation values for emotionally salient lyrics such as "*love*", which is expressed in heart-shaped motions. Furthermore, using our relationship detection method, we propose a method for retrieving dance motions from lyrics that outperforms previous text-to-motion retrieval methods, which focus on prose and non-dance motions.

1 Introduction

Platforms like YouTube and TikTok have popularized dance videos, allowing amateur dancers to express their creativity, but choreographing these performances is challenging for those without professional training. Consequently, previous studies have focused on synthesizing or retrieving dance motions that match the rhythms and timbres of music (Zhu et al., 2024). However, choreographic design is influenced by both audio attributes and lyrics, as dancers often interpret lyrics literally, such as by making heart shapes to express "*love*". However, no empirical studies have quantitatively analyzed the relationships between lyrics and dance, and no studies have specifically focused on retrieving dance motions associated with lyrics. Quantitative analysis of lyrics and dance motions offers two main contributions. Academically, this analysis provides clear evidence of how lyrics and dance motions are related. Practically, it enables the development of systems that suggest dance motions based on user-input lyrics, offering choreographic ideas to support dancers, particularly amateurs, in creating or enhancing their choreography.

Previous studies have quantified the relationship between prose and corresponding non-dance gestures (e.g., "*A man walks in a quarter circle to the left*") (Yu et al., 2024; Horie et al., 2023; Petrovich et al., 2023; Tevet et al., 2022). These methods assume that text and motion are analyzed in units of sentences and sequences, respectively, and that all text-motion pairs are inherently related. However, these methods fall short for analyzing interactions between lyrics and dance motions, because those interactions are often localized to specific parts of songs rather than uniformly distributed across the entire song. For example, functional words like "*the*" or "*is*" may not correspond to any dance motion, while emotionally charged words like "*love*" might correlate with specific motions. To analyze lyrics and dance motions, we need a method that can detect which frames[1] have motions related to lyrics.

This study proposes a novel data-driven method to analyze and quantify the relationship between lyrics and dance motions as illustrated in Figure 1. We hypothesize that lyrics and dance motions that co-occur in different songs are related. For example, if "*jump*" is sung while a jumping motion is performed in multiple songs, the co-occurrence suggests a meaningful relationship. To quantify these relationships, we first transform lyrics and

[1] A 'frame' refers to a single pose within a sequence of dance motions, similar to a frame in a video. Dance motions are recorded at a specific frames-per-second rate.

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 7901–7916

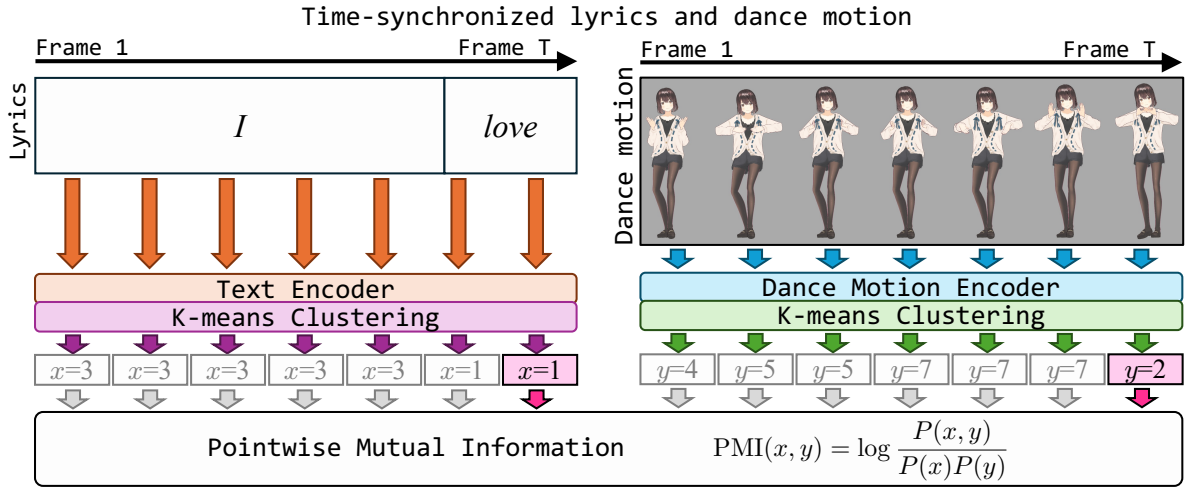

Figure 1: Overview of the proposed data-driven method.

dance motions of each frame into discrete symbols by using clustering methods. These symbols allow us to calculate their co-occurrence frequency. We then calculate Pointwise Mutual Information (PMI), a measure used in natural language processing (NLP) methods, and use it to evaluate these frequencies. By using PMI we find positive correlations for frequent co-occurrences and negative correlations for independent occurrences, thus detecting specific frames where lyrics and dance motions are related.

To investigate the effectiveness of our method, we have prepared a novel dataset of time-synchronized lyrics and dance motions, including the dancer's finger motions. Applying our method to this dataset, we have detected frames with high PMI values between lyrics and dance motions. For example, the lyric "*love*" corresponds to dance motions that form heart shapes with the hands, and the phrase "*getting lost*" is associated with motions that suggest searching for a destination. Our proposed method therefore makes it possible to quantify and analyze specific relationships between lyrics and dance motions, revealing valuable correlations.

In addition, we have developed a lyrics-to-dance motion retrieval method by utilizing our method to detect relationships between lyrics and dance motions. This method outperforms an existing text-to-motion retrieval method based on contrastive learning, which assumes that all text-motion pairs are inherently related. Unlike this general method that focuses on text and non-dance motions, our method specifically targets lyrics and dance motions that frequently co-occur across multiple songs, indicating a meaningful relationship. Our method overcomes the limitations observed in broader text-to-motion studies and improves retrieval performance, providing an effective solution in this area.

2 Related Work

2.1 Audio-to-Dance Motion Synthesis

Previous studies have focused on correlating dance motions with the rhythmic and timbral features of audio signals (Zhu et al., 2024). A notable trend in this area is the use of diffusion models to synthesize dance motions from music audio (Tseng et al., 2023; Dabral et al., 2023; Li et al., 2024; Zhang et al., 2024; Luo et al., 2024; Qi et al., 2023). These models represent dance motion as matrices defined by frame counts T and joint parameters J, and they use architectures such as U-Nets (Ronneberger et al., 2015) or Transformers (Vaswani et al., 2017) to reconstruct these matrices. While promising, synthesis based on diffusion models often results in long computation times and can produce motions plagued by unnatural artifacts such as jitter or sliding feet.

An alternative method involves motion graph-based synthesis (Chen et al., 2021; Au et al., 2022; Gao et al., 2022) structured into three phases: segmenting audio into musical bars, retrieving dance motions from a database for each segment, and ensuring natural transitions between sequences. This method, ideal for music with repetitive structures like verse-bridge-chorus, maintains consistency across similar musical sections.

Unlike these studies, we do not focus on synthesizing dance motions from audio signals. Instead, we present a novel method that quantifies the rela-

tionship between lyrics and specific dance motions, offering a unique approach within the field of dance motion analysis. Our method enriches the understanding of how lyrics influence and correspond to dance, and it provides findings that could improve future methods of dance motion synthesis and retrieval by ensuring that dance motions are more closely aligned with lyrical content.

2.2 Text and Motion Relationships

The relationship between prose text descriptions and corresponding non-dance motion has attracted considerable interest. Recent advances have used contrastive learning to embed text and motion feature vectors in a shared vector space, effectively aligning related text and motion vectors to improve the accuracy of motion vector retrieval (Yu et al., 2024; Horie et al., 2023; Petrovich et al., 2023; Tevet et al., 2022). In addition, the integration of large language models has extended the capabilities of text-motion analysis (Jiang et al., 2023). A particularly innovative method involves the use of vector quantized-variational auto-encoders to transform motion data into codebooks, which are then treated as pseudowords. These pseudowords are integrated with text to train language models, supporting a range of applications from text-to-motion synthesis to motion description generation and predictive modeling.

However, many existing methods assume a uniform relationship across all text-motion pairs, an assumption that may not reflect the complexity of lyrics-dance interactions. Contrary to the commonly assumed uniformity, lyrics-dance relationships may appear selectively, becoming prominent only in particular contexts, such as when emotionally charged words like "*love*" inspire certain dance motions. Our study examines how lyrics influence dance in different situations, allowing us to detail the multifaceted nature of these interactions.

3 Time-Synchronized Lyrics and Dance Motion Pair Data

Our goal is to analyze and quantify the relationship between lyrics and dance motions. To achieve this, we need data where each frame of dance motion is associated with specific words or sentences from the corresponding lyrics.

3.1 Data Collection

We collected 1,000 dance motion datasets (totaling 55.3 hours) from the MikuMikuDance community, where creators manually trace dance motions from dance videos on platforms such as YouTube and NicoNico or use motion capture technologies to create dance sequences. Our collection includes 979 traced motions and 21 captured motions. These dance motions correspond to 868 unique songs, as some songs have multiple associated dance motions.

We obtained the corresponding audio and lyrics for these songs from various online resources. Most of the lyrics are in Japanese, with some in English. To ensure synchronization between the dance motions and the audio, we manually aligned their start times and annotated the start and end times of each lyric sentence during audio playback. Word-level timing annotation, which is labor-intensive, was refined using an automatic synchronization method (Nakano and Goto, 2016) that allows precise alignment of individual words. As a result, each frame of the motion data, recorded at 30 frames per second, is associated with specific words and sentences. Frames without corresponding lyrics were assigned a padding token, represented as [PAD], to maintain sequence consistency. As these choreographies and lyrics are copyrighted, we do not plan to make the collected data publicly available. However, the code for training the proposed model and conducting the evaluation is available at `https://github.com/KentoW/lyrics-and-dance`.

3.2 Dance Motion Data Pre-processing

The human skeletal model in our dataset comprises 53 joints, each represented by global coordinates (x, y, z) and Euler angles (roll, pitch, yaw). See Appendix B for details on the structure of the human skeletal model. To avoid issues like gimbal lock, we convert these angles into a six-dimensional format using sine and cosine transformations: sin(roll), cos(roll), sin(pitch), cos(pitch), sin(yaw), and cos(yaw). Preliminary tests showed this transformation method to be more effective than using quaternions.

In this study, we define the unit of analysis for dance motions as bars for easier analysis. Using the downbeat tracking method (Böck et al., 2016), we segmented the dance motion and audio data into bars, discarding any bars shorter than one second. This resulted in a dataset of 119,691 bars: 92,723 with lyrics and 26,968 without.

To ensure uniform spatial positioning for consistent analysis, we adjusted the y axis to set the

minimum coordinate of the "*toe*" joints to zero and adjusted the x and z axes to align the human model's average position with the origin for each bar. We calculated velocities and accelerations for each joint based on its positional and angular data, including local xyz directional velocities and accelerations in six dimensions and point-to-point global velocities and accelerations in two dimensions. We derived the first and second derivatives of the six-dimensional angular parameters, integrating twelve additional dimensions. As a result, each joint frame contained 29-dimensional parameters, providing a detailed framework for comprehensive motion analysis. Finally, we normalized the 29-dimensional parameters to ensure that they ranged between -1 and 1.

4 Analyzing Lyrics and Dance Motion Relationships

In this section we present a method for detecting frames that demonstrate meaningful relationships between lyrics and dance motions within our time-synchronized data. Based on our hypothesis that the frequent co-occurrence of lyrics and dance motions across multiple songs indicates meaningful relationships, we use co-occurrence frequencies to quantitatively analyze these interactions.

To analyze the co-occurrence relationships at each frame, we utilize PMI, a metric used in NLP. PMI is calculated with the formula

$$\mathrm{PMI}(x, y) = \log \frac{P(x, y)}{P(x)P(y)} \qquad (1)$$

where x and y respectively represent the lyrics and dance motions at a specific frame. $P(x, y)$ is the probability of their co-occurrence within the same frame, and $P(x)$ and $P(y)$ are the probabilities of observing x and y independently at any frame. High PMI values at a frame indicate a strong association between the lyrics and dance motions at that particular moment.

To apply PMI, we convert lyrics and dance motions into discrete symbols x and y for each frame. This involves transforming feature vectors, derived from deep learning models, into a form suitable for PMI analysis. Using a clustering method, we categorize similar motions and lyrical expressions into codebooks x and y, which serve as the basis for calculating PMI, enabling us to detect frames where lyrics and dance motions are closely related.

The process begins with extracting feature vectors for lyrics and dance motions by using deep learning models. We then implement clustering to group similar lyrics and dance motions, facilitating the PMI calculations for our analysis.

4.1 Lyrics Feature Extraction

Before clustering lyrics, we calculate feature vectors using a language model, either at the word level or the sentence level. This distinction is crucial, as the relationship between lyrics and dance motions can vary. For example, "*jump*" may correspond to a specific motion, while "*I feel free like a bird in the sky*" can inspire broader, fluid motions, capturing the overall feeling.

To explore these relationships, we use a pre-trained multilingual Sentence-BERT model[2] (Reimers and Gurevych, 2019) to generate both word-level and sentence-level feature vectors. This method allows us to analyze how individual words and broader thematic content influence corresponding dance motions.

4.2 Dance Motion Feature Extraction

We developed a novel dance motion encoder to extract features from our dataset, which includes detailed elements like finger joint configurations and variable sequence lengths. This encoder preprocesses both the physical and expressive components of dance, providing features for clustering.

The encoder processes two data types per frame: motion sequences and affective features. Each frame's motion data, denoted S_t, is a matrix with dimensions $J \times P$, where J represents the number of joints and P represents the number of parameters per joint. These matrices, $S_1, \ldots, S_t, \ldots, S_T$, cover T frames within a musical bar, capturing detailed motion across the sequence.

Additionally, we include affective features for each frame, denoted as $\boldsymbol{a}_t$. These vectors, derived from geometric properties like volume, area, length, and curvature between joints, represent the expressive qualities of dance. The sequence of these features across a bar is denoted as $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_t, \ldots, \boldsymbol{a}_T$. Affective features recognized in human motion analysis (Kleinsmith and Bianchi-Berthouze, 2013; Crenn et al., 2016; Bhattacharya et al., 2020a,b, 2021) help capture the expressive aspects of dance, enriching our encoder's data input. By combining motion matrices and affective features, our encoder processes inputs that reflect both the physical execution and expressive dynamics of dance.

[2] https://huggingface.co/sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2

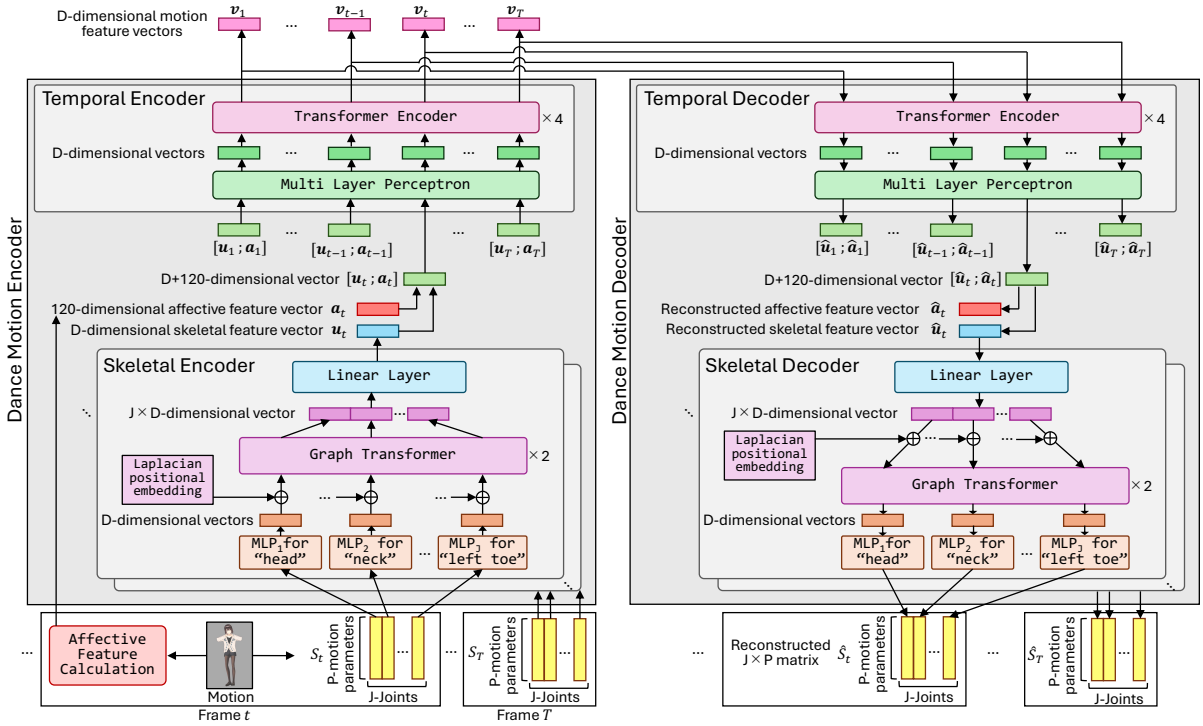

Figure 2: Overview of the proposed dance motion encoder and auto-encoder.

4.2.1 Affective Feature

We use 40 affective features, including the area of triangles formed by major joints, volume, distances between key joints, and limb curvature. These features provide a multidimensional view of the dancer's expressive state, revealing various emotional states. For example, a smaller area may suggest a reserved or tense posture, while a larger area implies openness and relaxation. See Appendix B for more details about the 40 affective features.

For each frame of motion data, we calculate these features along with their velocity and acceleration, forming a 120-dimensional affective feature vector. This method captures both the static posture and the dynamics of movement, which are crucial for understanding the fluidity and intensity of dance motions. We apply min-max normalization to these vectors, ensuring all features range from 0 to 1. This standardization allows for consistent and meaningful comparisons across different dance sequences.

4.2.2 Dance Motion Encoder Architecture

Our dance motion encoder consists of a skeletal encoder and a temporal encoder, as shown on the left side of Figure 2. The process starts with skeletal data for each frame S_t. Each P-dimensional joint vector in S_t is transformed by a specific Multi-Layer Perceptron (MLP) for each joint. These transformed vectors are fed into a Graph Transformer, which uses Laplacian positional embeddings to maintain the relative positions of the joints (Rao and Miao, 2023). The output is a $J \times D$-dimensional vector for the joints, which is flattened and compressed through a linear layer to produce a single D-dimensional vector representing the skeletal features for each frame, denoted as $\boldsymbol{u}_t$. This vector captures the comprehensive skeletal structure of the dance motions at each frame.

Simultaneously, affective features ($\boldsymbol{a}_t$) are calculated for each frame, capturing emotional dynamics through metrics like joint areas and volumes. Before inputting into the temporal encoder, skeletal feature vectors ($\boldsymbol{u}_t$) and affective features ($\boldsymbol{a}_t$) are concatenated, forming a combined $D + 120$-dimensional vector for each frame. This vector is then compressed through an MLP to ensure uniform D-dimensional vector consistency across all frames. The compressed vectors are fed into a Transformer encoder without positional embeddings (Haviv et al., 2022), emphasizing intrinsic interactions across frames rather than chronological order. This method generates a sequence of motion feature vectors ($[\boldsymbol{v}_1, \ldots, \boldsymbol{v}_t, \ldots, \boldsymbol{v}_T]$), which comprehensively represent both the skeletal structure and emotional dynamics of the dance.

To train our dance motion encoder, we use an auto-encoder method suitable for our dataset that lacks gold labels. The dance motion decoder, shown on the right side of Figure 2, is designed as a reverse architecture of the encoder. Both the encoder and the decoder are trained by minimizing the Mean Squared Error (MSE) between the inputs and reconstructed outputs. The training process reconstructs skeletal matrices $\hat{S}_t$ with tanh activation and affective features $\hat{\boldsymbol{a}}_t$ with sigmoid activation, ensuring the output matches the input:

$$Loss = \sum_{t=1}^{T} \Big(\mathrm{MSE}(S_t, \tanh(\hat{S}_t)) + \mathrm{MSE}(\boldsymbol{a}_t, \mathrm{sigmoid}(\hat{\boldsymbol{a}}_t)) \Big). \tag{2}$$

We train our auto-encoder on the full dataset of 119,691 bars using the AdamW optimizer (Loshchilov and Hutter, 2019), with a mini-batch size of 8 over 200 epochs. Our skeletal encoder processes data from 53 joints ($J = 53$) and 29 motion parameters per joint ($P = 29$). The encoder and decoder, with a dimensionality of $D = 256$, use 4 multi-heads and 2 layers for skeletal processing, and 8 multi-heads and 4 layers for temporal processing.

4.3 Lyrics and Dance Motion Clustering

To confirm that lyrics and dance motions that co-occur in multiple songs indicate a meaningful relationship, it is crucial to preprocess the data before applying k-means clustering. This is because repetitive elements within a single song could skew our analysis. For example, if a word like "*flower*" is associated with a particular dance motion multiple times within only a single song, their association does not necessarily indicate a meaningful relationship. If the word and motion co-occur in multiple songs, however, their co-occurrence is likely to be meaningful.

To avoid clustering bias caused by duplicate entries within our dataset, we preprocess both lyrics and dance motions before clustering. Specifically, we identify and unify duplicate sentences within the lyrics to ensure that each unique sentence is represented only once in the clustering process. This prevents the formation of clusters dominated by repeated sentences. Similarly, for dance motions, we treat vectors with a cosine similarity of 0.99 or higher as duplicates and unify them. These preprocessing steps enable more accurate clustering by ensuring both lyrics and dance motions are represented by distinct, non-redundant data points. With the data cleaned, we transform word vectors into the codebook x_w and sentence vectors into the codebook x_s for each frame. Similarly, dance motion vectors for each frame are transformed into the codebook y.

4.4 PMI Calculation

To calculate PMI, we adjust the standard approach to fit our hypothesis that lyrics and dance motions that co-occur in multiple songs are indeed related. We calculate the probabilities based on the number of songs in which lyrics and dance motions co-occur by using the following formulas:

$$P(x, y) = \frac{\#(x, y)}{\text{total number of songs}}, \tag{3}$$

$$P(x) = \frac{\#(x)}{\text{total number of songs}}, \tag{4}$$

$$P(y) = \frac{\#(y)}{\text{total number of songs}}. \tag{5}$$

Here $\#(x, y)$ indicates the number of songs where x and y co-occur within the same frame, while $\#(x)$ and $\#(y)$ indicate the numbers of songs where x and y appear independently. Equations (3), (4) and (5) are then used to calculate PMI as defined in Equation (1), emphasizing the importance of frequent co-occurrences in different songs. Additionally, to avoid bias from single occurrences ($\#(x, y) = 1$), which can misleadingly suggest strong relationships, we set PMI for these cases to zero. For clearer interpretation, we use Normalized PMI (NPMI) by normalizing PMI values to a scale between -1 and 1.

4.5 Analysis Setup

To comprehensively examine specific correlations between lyrics and dance motions across our entire dataset, we utilize high NPMI values in all lyric-motion pairs without dividing the dataset into training and test sets. We analyze these pairs using two types of lyric codebooks, x_w for word-level and x_s for sentence-level lyrics, alongside one codebook, y, for dance motions. To obtain each codebook, we applied k-means clustering to each set of lyric or motion vectors across varying codebook sizes from 500 to 7000, in increments of 500, to explore different levels of granularity and find the most effective categorization for capturing nuanced relationships.

For each codebook size combination, we calculated NPMI values between all cluster pairs, one

Table 1: Numbers and percentages of songs, bars, and frames with positive NPMI values.

Entity	In the case of NPMI(x_w, y)	In the case of NPMI(x_s, y)
Number of songs	781 songs (78.10%)	980 songs (98.00%)
Number of bars	2,671 bars (2.88%)	16,161 bars (17.43%)
Number of frames	14,431 frames (0.03%)	91,593 frames (1.99%)

from the motion codebook and one from the lyrics codebook. This helped us detect the optimal granularity that maximizes the meaningful mutual information. Specifically, our highest NPMI values were 0.82 for word-level analysis (with a codebook size of 5000 for lyrics and 4500 for motions) and 0.93 for sentence-level analysis (with a codebook size of 6500 for both lyrics and motions). We use these codebook sizes for the following analysis.

4.6 Analysis Results

Table 1 shows that while the majority of songs (78.1% for word-level analysis and 98.0% for sentence-level analysis) have positive NPMI values, indicating strong lyrics-motion relationships, the numbers of bars and frames with positive NPMI values are quite small. This suggests that meaningful interactions between lyrics and dance motions are localized to specific parts of songs, supporting our hypothesis that meaningful relationships, while present, are not uniformly distributed across songs.

The examples shown in Figure 3 were selected from musical bars with positive NPMI values to illustrate specific relationships as we interpreted them. For example, emotional expressions in lyrics, such as "*love*", are often translated into heart-shaped gestures in dance. Sentence-level correlations show clear patterns, such as "*getting lost*" with a peering motion, demonstrating how broader narrative elements within lyrics can influence dance motions. See Appendix D for other examples of interpretable relationships between lyrics and corresponding dance motions. This analysis confirms that our data-driven method can uncover intuitive relationships between lyrics and dance motions.

5 Lyrics-to-Dance Motion Retrieval

We developed a method that allows input of a single musical bar of lyrics to retrieve the corresponding bar of dance motions. The input is a sequence of lyric words within a musical bar. The retrieved output is a ranked list of musical bars containing the corresponding motions.

For the retrieval task, we use Dynamic Time Warping (DTW) (Berndt and Clifford, 1994) to measure the similarity between the input lyrics and available dance motions. DTW is ideal for handling time series data of varying lengths and for capturing partial frame similarities, thereby accounting for how lyrics and dance motions relate within a bar.

In our DTW implementation, we derive the cost matrix from the Normalized Pointwise Mutual Information (NPMI) between the lyric and dance motion codebooks. The substitution (match) cost between elements x and y is defined as

$$\text{cost}(x, y) = 1 - \big(1 + \text{NPMI}(x, y)\big)/2. \quad (6)$$

This formula assigns lower costs when there is a stronger relationship between x and y (i.e., higher NPMI values) and assigns higher costs when the relationship is weaker. For insertions and deletions, we assign a fixed cost of 1. This means that when aligning sequences, inserting or deleting an element incurs a constant penalty, regardless of the specific elements involved. This use of a fixed cost was based on preliminary experiments showing that using variable costs degraded performance in the retrieval task.

We implemented two retrieval strategies: word-to-dance motion retrieval (W2D) based on the word-to-motion NPMI(x_w, y) and sentence-to-dance motion retrieval (S2D) based on the sentence-to-motion NPMI(x_s, y).

In this section, our dataset comprised 868 songs and was divided into 78,875 bars (85%) for training, 5,237 bars (5%) for development, and 8,611 bars (10%) for testing. Unlike Section 4 where models were trained on the full dataset, here we specifically trained the motion encoder and decoder, and conducted k-means clustering from scratch using only the training subset. In training the motion encoder, we utilized early stopping based on the development subset, ensuring precision in model tuning. We confined k-means clustering and NPMI calculations strictly to the training data, guaranteeing that these models were accurately calibrated for the specific tasks described in this section. Additionally, we adjusted the codebook sizes for k-means clustering, with optimal sizes of 6000 for lyrics and 7000 for motions in the W2D method, and 3500 for lyrics and 7000 for motions in the S2D method.

As there are no existing methods specifically designed for lyrics-to-dance motion retrieval, to eval-

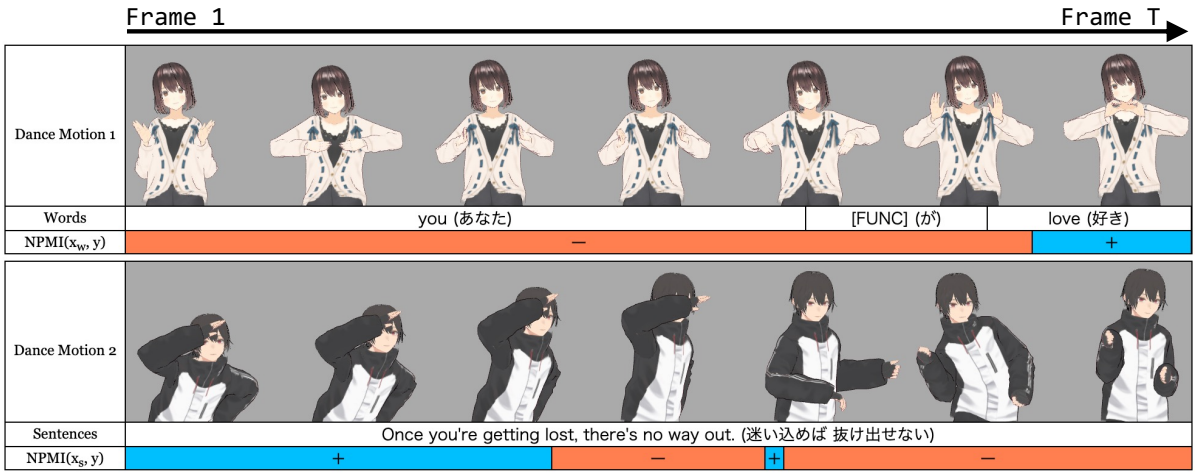

Figure 3: Examples of lyric and dance motion relationships from bars with positive NPMI values. The two examples include one that shows word-level correlations and one that shows sentence-level correlations. Each example pairs a synchronized lyric translated into English (with the original Japanese in parentheses) with its corresponding dance motion. The term "FUNC" indicates Japanese functional words that defy easy translation. NPMI values are indicated by visual cues: negative values are indicated by a minus sign in an orange box, positive values by a plus sign in a blue box, and zero values by 0.

Table 2: Lyrics-to-dance motion retrieval comparison.

Method	MRR ↑	1/MRR ↓
Random	0.00113	884
Contrastive learning	0.00151	663
Proposed method (W2D)	**0.01905**	**53**
Proposed method (S2D)	**0.01837**	**54**

uate our lyrics-to-dance motion retrieval method we devised a baseline using contrastive learning techniques adapted from text-to-motion retrieval studies (Yu et al., 2024; Horie et al., 2023; Petrovich et al., 2023; Tevet et al., 2022). Additionally, we included a basic random selection method that randomly selects dance motions from the test set. For the contrastive learning method, sentence vectors from a bar, generated by pre-trained Sentence-BERT, were first averaged into a 384-dimensional vector and then compressed to a 256-dimensional vector using an MLP. Dance motion feature vectors from our pre-trained motion encoder were similarly processed into a sequence of 256-dimensional vectors by an MLP. Those vectors were then averaged into a single 256-dimensional vector via mean pooling. We fixed the parameters for Sentence-BERT and the motion encoder, focusing training on the MLP parameters using a contrastive loss function to distinguish between matching and non-matching pairs. Training employed the AdamW optimizer with a mini-batch size of 16 over 200 epochs, with early stopping triggered after 10 epochs without improvement in development loss.

The experimental results presented in Table 2 show that while the contrastive learning method slightly outperforms the random method, both of our proposed methods achieve improvements, with Mean Reciprocal Ranks (MRRs) of 0.019 for W2D and 0.018 for S2D. This performance indicates that for a total of 8,611 bars, W2D ranks the correct dance motions within the top 53 positions on average, while S2D ranks them within the top 54 positions. A statistical t-test between W2D and S2D results yields a p-value of 0.480, indicating no significant difference between the two methods. These results support our hypothesis that while not all lyrics and dance motions share inherent relationships, there are meaningful relationships in certain instances.

6 Conclusion and Future Work

This paper introduces a novel method for quantifying the relationship between lyrics and dance motions that uses co-occurrence frequency. Our method effectively detects where lyrics correlate with dance motions and was validated by identifying meaningful relationships, such as the association between "*love*" and heart-shaped motions, and outperforming a previous method in a lyrics-to-dance motion retrieval task.

The success of this method not only impacts fields such as lyrics information processing, dance

information processing, music information retrieval, and computer vision but also opens promising avenues for interdisciplinary studies and enhances the integration of text content into choreographic design. Future work will aim to integrate our method into existing audio-to-dance motion retrieval methods to improve their accuracy in matching dance motions with lyrics.

7 Limitations

First, the dataset we used to evaluate our method's performance contained predominantly Japanese songs and thus may not represent the global musical landscape. Although our approach is adaptable, its generalizability needs to be validated with different linguistic inputs in the future. The dance motions in our dataset were mainly in the style of Japanese popular music and lacked the diversity of dance styles such as breakdance or street dance. Nonetheless, our method can theoretically be applied to other styles.

Second, while our method successfully identifies many significant lyric-dance associations, not all relationships are easily interpretable. This limitation highlights the challenges of using purely data-driven approaches without additional contextual or cultural insights. Additionally, due to computational constraints, we used the multilingual Sentence-BERT for lyric analysis, forgoing more advanced large-scale language models that might enhance our method's performance. Our study paved the way for future research using such computationally intensive models.

8 Acknowledgments

This work was supported in part by JST CREST Grant Number JPMJCR20D4, Japan.

References

Ho Yin Au, Jie Chen, Junkun Jiang, and Yike Guo. 2022. ChoreoGraph: Music-conditioned automatic dance choreography over a style and tempo consistent dynamic graph. In *The 30th ACM International Conference on Multimedia, MM 2022*, pages 3917–3925.

Donald J Berndt and James Clifford. 1994. Using dynamic time warping to find patterns in time series. In *Proceedings of Workshop on Knowledge Discovery in Databases*, pages 359–370.

Uttaran Bhattacharya, Trisha Mittal, Rohan Chandra, Tanmay Randhavane, Aniket Bera, and Dinesh Manocha. 2020a. STEP: Spatial temporal graph convolutional networks for emotion perception from gaits. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020*, pages 1342–1350.

Uttaran Bhattacharya, Nicholas Rewkowski, Abhishek Banerjee, Pooja Guhan, Aniket Bera, and Dinesh Manocha. 2021. Text2Gestures: A transformer-based network for generating emotive body gestures for virtual agents. In *IEEE Virtual Reality and 3D User Interfaces, VR 2021*, pages 160–169.

Uttaran Bhattacharya, Nicholas Rewkowski, Pooja Guhan, Niall L. Williams, Trisha Mittal, Aniket Bera, and Dinesh Manocha. 2020b. Generating emotive gaits for virtual agents using affect-based autoregression. In *2020 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2020*, pages 24–35.

Sebastian Böck, Florian Krebs, and Gerhard Widmer. 2016. Joint beat and downbeat tracking with recurrent neural networks. In *Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR 2016*, pages 255–261.

Kang Chen, Zhipeng Tan, Jin Lei, Song-Hai Zhang, Yuan-Chen Guo, Weidong Zhang, and Shi-Min Hu. 2021. ChoreoMaster: Choreography-oriented music-driven dance synthesis. *ACM Trans. Graph.*, 40(4):145:1–145:13.

Arthur Crenn, Rizwan Ahmed Khan, Alexandre Meyer, and Saïda Bouakaz. 2016. Body expression recognition from animated 3D skeleton. In *International Conference on 3D Imaging, IC3D 2016*, pages 1–7.

Rishabh Dabral, Muhammad Hamza Mughal, Vladislav Golyanik, and Christian Theobalt. 2023. MoFusion: A framework for denoising-diffusion-based motion synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023*, pages 9760–9770.

Jibin Gao, Junfu Pu, Honglun Zhang, Ying Shan, and Wei-Shi Zheng. 2022. PC-Dance: Posture-controllable music-driven dance synthesis. In *The 30th ACM International Conference on Multimedia, MM 2022*, pages 1261–1269.

Adi Haviv, Ori Ram, Ofir Press, Peter Izsak, and Omer Levy. 2022. Transformer language models without positional encodings still learn positional information. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1382–1390.

Junpei Horie, Wataru Noguchi, Hiroyuki Iizuka, and Masahito Yamamoto. 2023. Learning shared embedding representation of motion and text using contrastive learning. *Artif. Life Robotics*, 28(1):148–157.

Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. 2023. MotionGPT: Human motion as a foreign language. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023*.

Andrea Kleinsmith and Nadia Bianchi-Berthouze. 2013. Affective body expression perception and recognition: A survey. *IEEE Trans. Affect. Comput.*, 4(1):15–33.

Ronghui Li, Yuxiang Zhang, Yachao Zhang, Hongwen Zhang, Jie Guo, Yan Zhang, Yebin Liu, and Xiu Li. 2024. Lodge: A coarse to fine diffusion network for long dance generation guided by the characteristic dance primitives. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024*, pages 1524–1534.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of the 7th International Conference on Learning Representations, ICLR 2019*.

Zhenye Luo, Min Ren, Xuecai Hu, Yongzhen Huang, and Li Yao. 2024. POPDG: Popular 3D dance generation with PopDanceSet. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024*, pages 26984–26993.

Tomoyasu Nakano and Masataka Goto. 2016. LyricListPlayer: A consecutive-query-by-playback interface for retrieving similar word sequences from different song lyrics. In *Proceedings of the 13th Sound and Music Computing Conference, SMC 2016*, pages 344–349.

Mathis Petrovich, Michael J. Black, and Gül Varol. 2023. TMR: Text-to-motion retrieval using contrastive 3D human motion synthesis. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023*, pages 9454–9463.

Qiaosong Qi, Le Zhuo, Aixi Zhang, Yue Liao, Fei Fang, Si Liu, and Shuicheng Yan. 2023. DiffDance: Cascaded human motion diffusion model for dance generation. In *Proceedings of the 31st ACM International Conference on Multimedia, MM 2023*, pages 1374–1382.

Haocong Rao and Chunyan Miao. 2023. TranSG: Transformer-based skeleton graph prototype contrastive learning with structure-trajectory prompted reconstruction for person re-identification. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023*, pages 22118–22128.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019*, pages 3980–3990.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention, MICCAI 2015*, volume 9351, pages 234–241.

Guy Tevet, Brian Gordon, Amir Hertz, Amit H. Bermano, and Daniel Cohen-Or. 2022. MotionCLIP: Exposing human motion generation to CLIP space. In *European Conference on Computer Vision, ECCV 2022*, volume 13682 of *Lecture Notes in Computer Science*, pages 358–374.

Jonathan Tseng, Rodrigo Castellon, and C. Karen Liu. 2023. EDGE: Editable dance generation from music. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023*, pages 448–458.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pages 5998–6008.

Qing Yu, Mikihiro Tanaka, and Kent Fujiwara. 2024. Exploring vision transformers for 3D human motion-language models with motion patches. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024*, pages 937–946.

Canyu Zhang, Youbao Tang, Ning Zhang, Ruei-Sung Lin, Mei Han, Jing Xiao, and Song Wang. 2024. Bidirectional autoregressive diffusion model for dance generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024*, pages 687–696.

Wentao Zhu, Xiaoxuan Ma, Dongwoo Ro, Hai Ci, Jinlu Zhang, Jiaxin Shi, Feng Gao, Qi Tian, and Yizhou Wang. 2024. Human motion generation: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(4):2430–2449.

A Implementation Details

The training of the dance motion encoder and decoder was implemented using PyTorch[3] and conducted on an NVIDIA V100 GPU. The total number of trainable parameters for the dance motion encoder-decoder is 5,858,221. For clustering, we employed the k-means algorithm implemented in scikit-learn[4], and statistical significance testing was performed using the t-test function from scipy[5]. Due to computational resource constraints, all results presented in this paper are based on a single run of the experiments.

B Human Skeletal Model and Affective Features

The left side of Figure 4 illustrates the human skeletal model with 53 joints, while the right side

[3] https://pytorch.org
[4] https://scikit-learn.org/stable/
[5] https://scipy.org

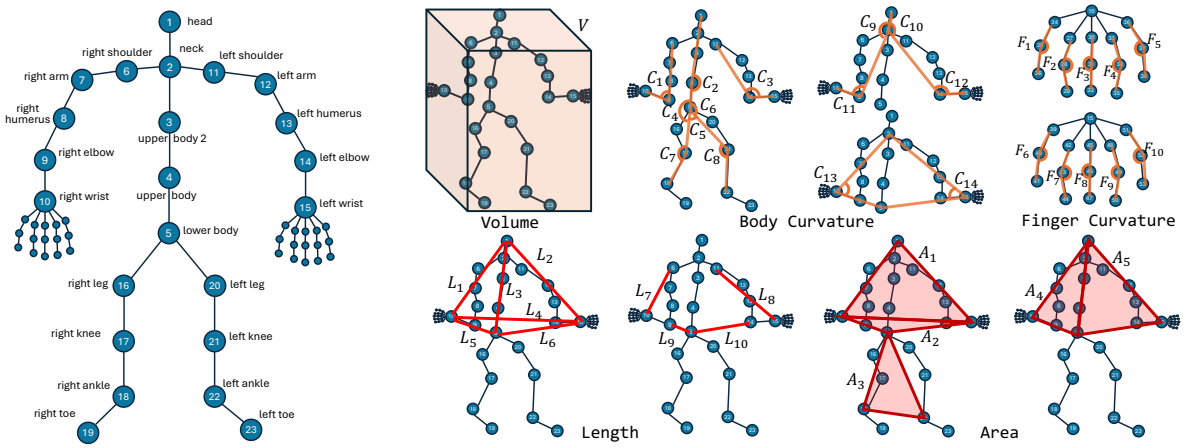

Figure 4: Detailed specifications of human skeletal model and affective features.

Table 3: Retrieval performance with skeletal and affective feature vectors. The codebook size in the table is displayed as a pair of values, with the first representing the codebook size for lyrics and the second representing the codebook size for motions.

Retrieval strategy	Feature vector	Codebook size	MRR ↑	1/MRR ↓
Word-to-Dance motion (W2D)	Both feature vectors	6000, 7000	**0.01905**	**53**
	Skeletal feature vectors	6000, 5000	0.01641	61
	Affective feature vectors	6000, 5000	0.01757	57
Sentence-to-Dance motion (S2D)	Both feature vectors	3500, 7000	**0.01837**	**54**
	Skeletal feature vectors	3000, 6500	0.01547	65
	Affective feature vectors	5000, 4500	0.01475	68

displays 40 affective features designed to express emotional states through body language. These features include volume (V) calculated from the distances between the extremal joint coordinates, length (L) measured across ten different joint pairs, area (A) derived from five types of triangles formed by triplets of joints, body joint curvature (C) calculated from 14 different measurements, and finger joint curvature (F) represented by ten different calculations that together quantify expressive body dynamics.

C Effectiveness of Combining Skeletal and Affective Feature Vectors

In this section, we investigate the contribution of skeletal feature vectors and affective feature vectors to the performance of our dance motion encoder through an ablation test.

For this experiment, we retrained the dance motion encoder using only skeletal feature vectors and, separately, using only affective feature vectors. The training parameters and configurations were kept identical to those used in the proposed method. Additionally, the optimal codebook sizes were determined separately for the W2D and S2D methods.

The results are shown in Table 3. The table shows that the model using both skeletal and affective feature vectors achieves better retrieval performance than models using either type of feature vector alone. This improvement is statistically significant, with p-values below 0.05 as determined by t-tests. These results suggest that the combination of skeletal and affective feature vectors effectively contributes to the lyrics-to-dance motion retrieval task.

D Additional Examples of Lyric-Dance Relationships

Figures 5, 6, 7, and 8 present additional examples of the relationships between lyrics and dance motions. Figures 5 and 6 present the relationships between lyric words and dance motions, while Figures 7 and 8 present the relationships between entire lyric sentences and dance motions. Table 4 lists the song titles and lyricists presented in Figures 3, 5, 6, 7, and 8.

In Figure 5, several relationships can be ob-

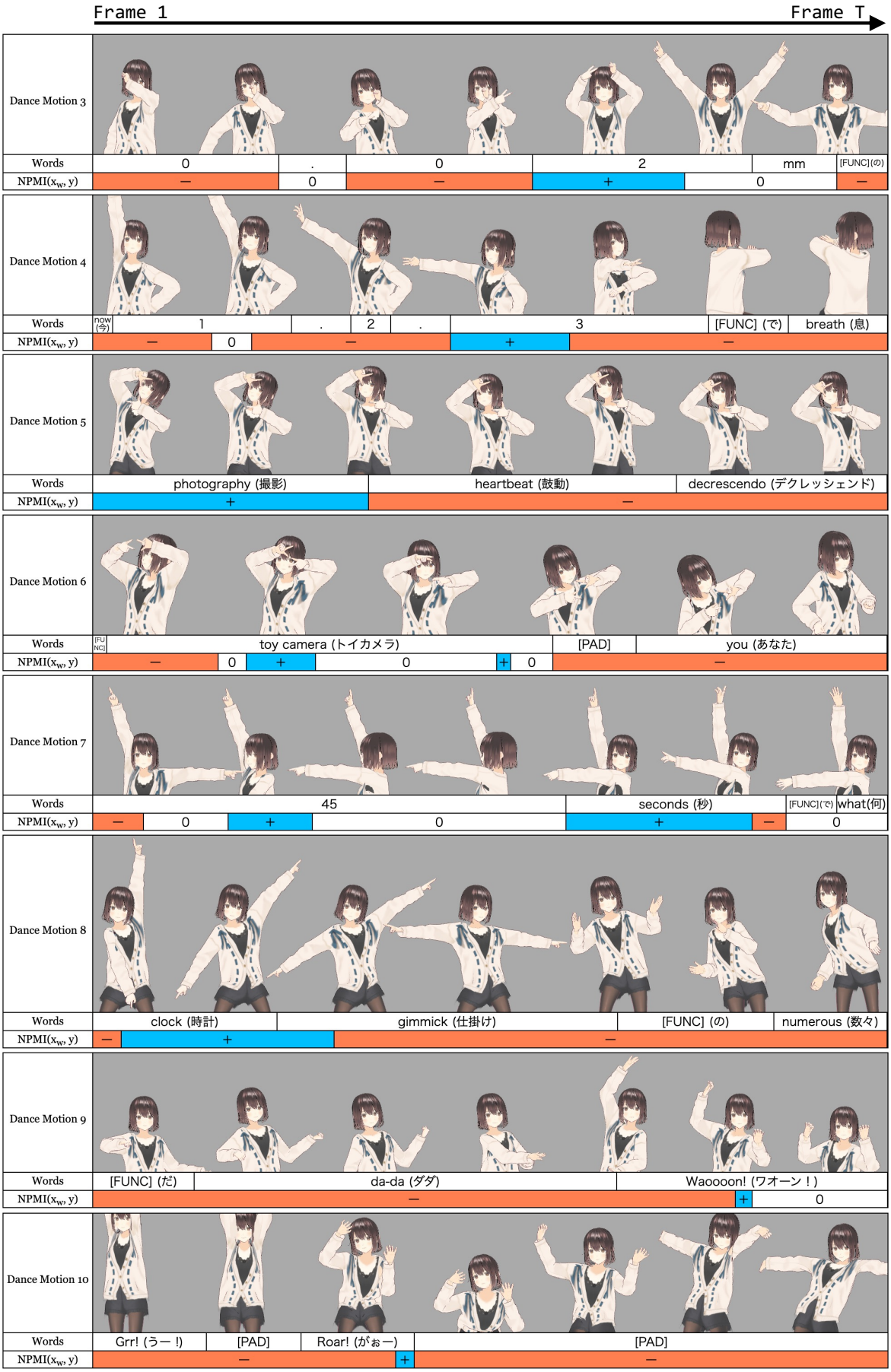

Figure 5: Examples of lyric and dance motion relationships with positive NPMI (Part 1/4).

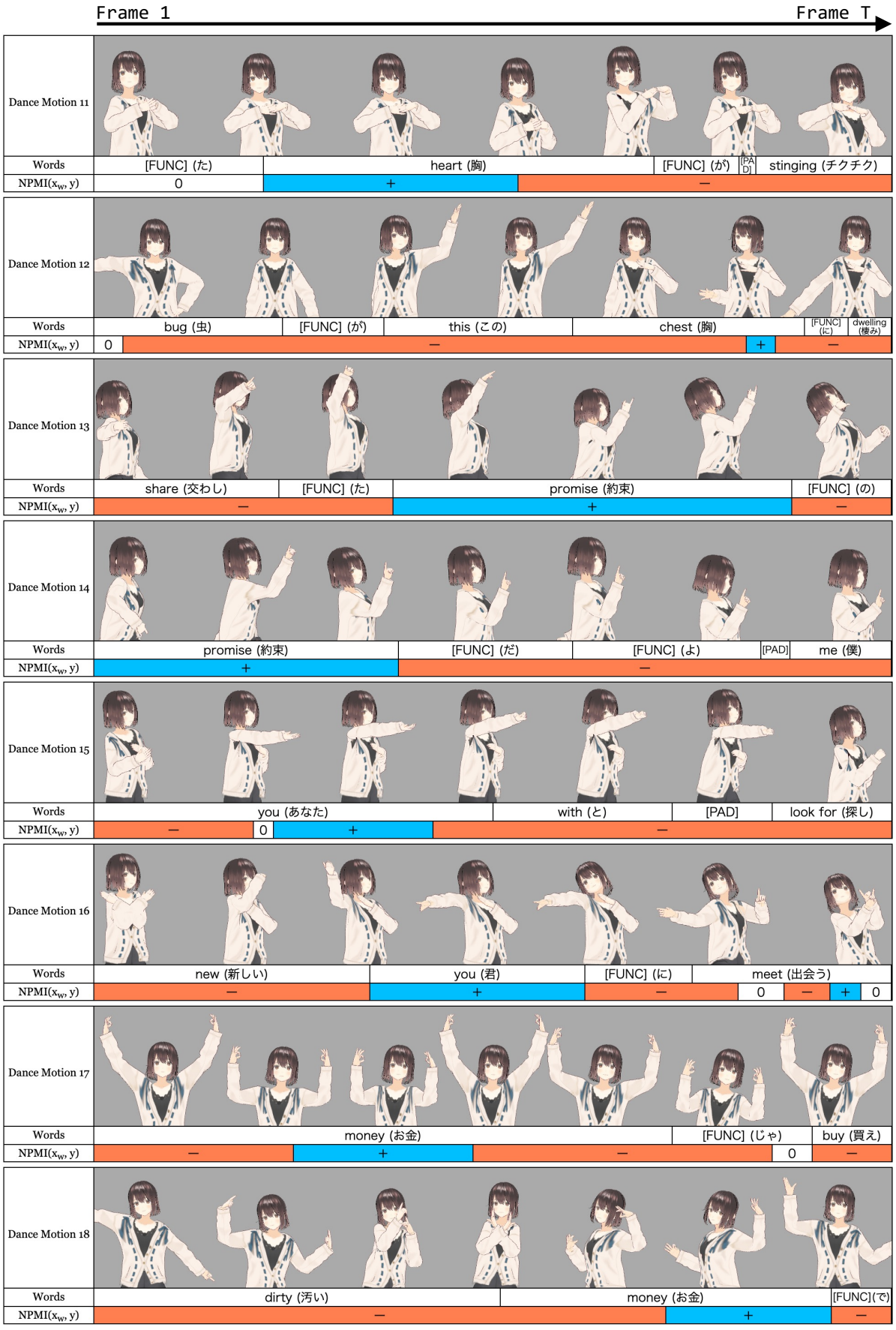

Figure 6: Examples of lyric and dance motion relationships with positive NPMI (Part 2/4).

Figure 7: Examples of lyric and dance motion relationships with positive NPMI (Part 3/4).

Figure 8: Examples of lyric and dance motion relationships with positive NPMI (Part 4/4).

served. Dance motions 3 and 4 correspond to numbers such as "*2*" and "*3*," where dancers use their fingers to count. Dance motions 5 and 6 show that words related to photography, such as "*photography*" and "*toy camera*," are associated with framing gestures using fingers. Dance motions 7 and 8 reveal that time-related words like "*seconds*" and "*clock*" correspond to motions where the arms mimic the hands of a clock. Dance motions 9 and 10 demonstrate that animal sounds such as "*Waoooon!*" and "*Roar!*" correspond to hand gestures resembling animal claws.

In Figure 6, when words such as "*heart*" or "*chest*" are sung, the corresponding dance motions 11 and 12 involve placing a hand on the chest. Dance motions 13 and 14 correspond to the word "*promise*," where the dancer raises their pinky finger, a gesture that in some cultures symbolizes a promise. Dance motions 15 and 16 correspond to the word "*you*," where the dancer extends an arm forward, pointing toward the audience. Dance motions 17 and 18 correspond to the word "*money*," where the dancer's finger form a circular shape representing a coin. These examples demonstrate how specific words in lyrics influence corresponding dance motions.

Figure 7 illustrates examples of the relationship between lyric sentences and dance motions. Dance motions 19 and 20 correspond to sentences related to "*new world*," where the dancer raises an index finger toward the sky, symbolizing the gesture of pointing to a new world. Dance motions 21 and 22 align with sentences expressing "*I meet you*," where the dancer uses an index finger to point toward "*you*." Dance motions 23 and 24 correspond to sentences about "*white snow*," with the dancer making a motion that mimics catching falling snow with their hands. Dance motions 25 and 26 correspond to sentences about "*selfies*," where the dancer makes a gesture of holding a camera while showing a peace sign, a common pose for taking photos in some cultures.

Figure 8 also presents examples of the relationship between lyric sentences and dance motions. Dance motions 27 and 28 correspond to negations using "*not*," where the dancer crosses their arms to form an "X," a gesture that signifies negation in some cultures. Dance motions 29 and 30 relate to sentences about losing something, with the dancer pointing to the ground with their index finger as if indicating a lost object. Dance motions 31 and 32 correspond to sentences containing the phrase "*I love*," where the dancer touches both cheeks with their hands in a cute expression. Dance motions 33 and 34 align with sentences related to love, where the dancer forms a heart shape with their hands. These examples demonstrate that our proposed method can discover various lyric-dance motion relationships in a data-driven manner.

Table 4: Song titles and lyricists referenced in Figures 3, 5, 6, 7, and 8.

Dance motion	**Song title / Lyricist**
1	エイリアンエイリアン / ナユタン星人
2	極楽浄土 / MARiA
3	聖槍爆裂ボーイ / れるりり・もじゃ
4	奇跡さえも / Omoi
5	シビュラ / wotaku
6	GIFT / 花束P
7	45秒 / れすぽん
8	Love Timer / emon (Tes.)
9	ルマ / かいりきベア
10	ようこそジャパリパークへ / 大石昌良
11	トモダチ以上のえと・せとら / 曲者P
12	プラリネ / かしこ。
13	Who? / Azari
14	晴天を穿つ / 傘村トータ
15	ダダダダ天使 / ナユタン星人
16	愛言葉Ⅲ / DECO*27
17	キラメキラリ / yura
18	妄想税 / DECO*27
19	Brand New World / 三日月美嘉
20	Melody Line / SmileR
21	リバースユニバース / ナユタン星人
22	Stocking Filler / nuru
23	Stocking Filler / nuru
24	Snow Fairy Story / 40mP
25	ダーリンダンス / かいりきベア
26	少女溶解 / 砂粒
27	キャットアイメイク / 奏音69
28	愛言葉Ⅳ / DECO*27
29	Fantastic Night / ぺぺろんP
30	トリノコシティ / 40mP
31	ぴんこすてぃっくLuv / れをる
32	おじゃま虫 / DECO*27
33	金星のダンス / ナユタン星人
34	チーズケーキクライシス / TOKOTOKO