MCLEMCD: Multi-modal Collaborative Learning Encoder for Enhanced Music Classification from Dances

Wenjuan Gong[1], Qingshuang Yu[1], Wendong Huang[1], Peng Cheng[2] and Jordi Gonzàlez[3]

[1]College of Computer Science and Technology, China University of Petroleum (East China), No. 66, Changjiangxi Road, Huangdao, 266555, China.

[2]Institute of High Performance Computing, A*STAR, 1 Fusionopolis Way, #16-16 Connexis (North Tower), Singapore.

[3]Computer Vision Center, Autonomous University of Barcelona, Edificio O, Campus UAB, Bellaterra (Cerdanyola), Barcelona, 08193, Spain.

Contributing authors: wenjuangong@upc.edu.cn; z19070034@s.upc.edu.cn; z21070220@s.upc.edu.cn; cheng_peng@ihpc.a-star.edu.sg; Jordi.Gonzalez@uab.cat;

Abstract

Music classification is widely applied in automatic organization of music archives, intelligent music interface, etc. Music is frequently accompanied by other media, such as image sequences. How to combine various types of media for various tasks is natural for human but extremely difficult for machines. In this work, we propose a collaborative learning method to combine dancing motions and music cues for music classification and apply it to music recommendation from dancing motion. Dancing motions in the form of 3D joint positions contain cyclic motions synchronized with music beats and a collabrative autoencoder is designed for fusing music cues into dancing motion feature extraction module. The proposed method achieved **98.07%** on MusicToDance dataset and **65.29%** on AIST++ dataset. The code to run all experiments is available at https://github.com/wenjgong/musicmotion.

Keywords: Multi-media processing, collaborative learning, music recommendation

1 Introduction

Music plays a crucial part in expressing emotions and artistic communications. Music classification is widely applied in various applications, such as, automatic organizing of music archives, and intelligent music interface [1, 2]. Automatic music classification [3, 4] can be further categorized into music genre classification [3–5]) and music emotion classification [6–8].

Even though music is usually accompanied by other media, such as, images or dancing motions, related works on music classification are mainly based on audio features (rhythm) [9] and their text descriptions (lyrics, audio tags, user comments, etc.) [10]. Other media inputs, such as dancing motions [11] that are synchronized with music beats [12], can also contribute. Collaborative learning from music-motion modalities are widely applied in human-computer interaction [13], robotics [14], auto choreography [15], etc. Most of the studies on collaborative learning from music and motion data concentrate on dancing motion creation from music [16–19], and music indexing using motion cues [20, 21]. In this work, we explored collaborative music classification from dancing motions, which were seldom studied in previous works.

Commonly used music features include Mel Frequency Cepstral Coefficients (MFCCs) [22], constant-Q chromagram [23], onset strength envelope [24], tempogram [25], etc. Each feature extracts a certain type of information. e.g., MFCCs transforms audio waveforms from temporal signal into frequency space and extracts features in frequency space, while tempogram extracts the start frame and music beats. Because these features extract information from different spaces, it is tricky to fuse them. MFCCs represents distribution of the energy of audio signals in different frequency ranges, and is widely used as inputs to deep learning based methods [26–28]. But the limitation of using one MFCCs feature is loss of tempo information [29], thus reducing task performance. Studies show that dancing motions are synchronized with their background music in tempo [30]. Dancing motions in form of 3D joint positions are unambiguous information [31], and they contain cyclic information synchronized with music beats. Beat-like cyclic information can be extracted from joint motions.

Based on the above studies and observations, we propose a collaborative learning method for a deep motion extractor and a music encoder to enhance music recommendation accuracy. A motion encoder and a music autoencoder extract motion and music features. Features of the two modalities are aligned using a collaborative learning, so that music classification performance

is improved. The music classification method is further used for music recommendation from dancing motions, where background music is recommended based on predicted motion/music categories.

The contributions of this work are as follows.

1. Collaborative learning is introduced in music classification. Music features are extracted using a encoder module, which are later mapped to their extracted motion features to learn correspondences between two types of media information.
2. Music classification accuracy on MuiscToDance dataset is enhanced by 6.77% using the proposed multi-modal collaborative learning method with a fully connection pair-wise network between music and motion features (98.07%) compared with method without collaborative learning [42] (91.3%).

2 Related Works

2.1 Correlations between Motions and Music

Dancing motions and their background music are emotionally correlated [11]. Furthermore, dancing motion cycles are synchronized with their background music beats [12]. Collaborative learning on music-motion modalities fuses multi-modal information effectively. One exemplar work on collaborative learning from music and motions was music beat feature extraction using motion sequences in videos [32], which denoted music beat features using motion trajectories and turnings in videos.

Motions in videos are ambiguous due to loss of depth information and constrains imposed by viewpoints [33]. As a result, we use 3D dancing motions in this work. They are unambiguous 3D joint positions, from which we are able to extract more accurate music beat information. Studies from Shi et al. [34] showed that music and dancing motions are closely correlated. So we explore collaborative learning from music and 3D dancing motions to enhance music classification (hence music recommendation) performance.

2.2 Multi-modal Collaborative Learning from Motions and Music

Collaborative learning methods are widely applied in statistical analysis [35] for enhanced visual detection [36] and visual-audio speech recognition [37]. Multi-modal collaborative learning not only improves performance of a task by incorporating multi-modal data, but also helps to ease the problem of lacking of data [38].

Multi-modal collaborative learning from music and motions are extensively studied. Some work aims for a better music representation. For example, co-aligned autoencoders for learning semantically enriched audio representations (Coala) [39] correlated audio contents and text labels through auto-encoder,

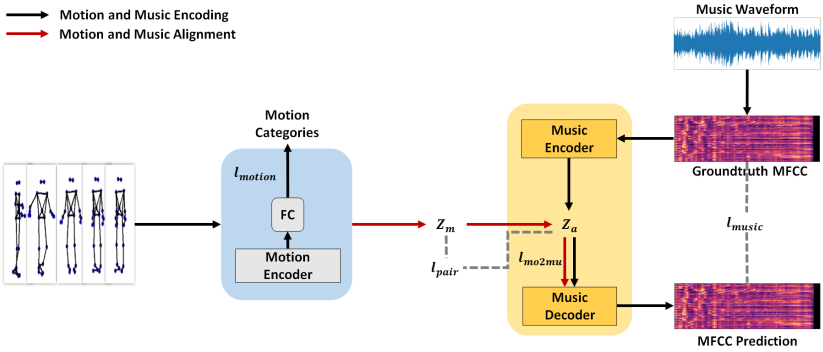

Fig. 1 Architecture of the proposed MCLEMCD model. The model is composed of two parts: a motion encoder and a music encoder. Motion features extracted using motion encoder are aligned with music features extracted by music auto-encoder through a collaborative learning module (between Z_m and Z_a). Black lines represent respective training routes for motions and music, and red lines represent collaborative training process of feature alignment.

denoted text labels of audio files using embedded feature from encoder, and reduced the costs of data labelling.

Others generate dancing motions from accompanying music. Deep learning based models, including GANs [40], causal convolution [14] and temporal networks [41] are utilized for this task. Lee et al. [40] proposed to generate dancing motions from music. They split dancing sequences into basic dancing units using auto-encoder, and composed a new dancing sequence using GANs so that generated dancing motions matched with music style and music beats. In test stage, style and beat information was extracted from music audio inputs, dancing units were generated according to style, and dancing sequence was finalized through composing dancing units according to music beats. Lee et al. [14] proposed auto choregraphy based on music styles. In order to learn correspondences between dancing and music sequences, they applied causal convolution to predict next motion frame based on dancing joint positions in current frame. And they incorporated dilated convolution to enlarge receptive field of encoder. During test, initial joint positions of start frame were sent into network and combined with audio encoder to generate joint positions of next frame. Predicted results were recursively input into the model to predict subsequent motion frames. Qi et al. [41], on the other hand, proposed a novel Seq2Seq framework to learn correlation between music denoted with MFCCs and dance denoted with key point coordinates detected using Openpose.

Collaborative music classification from dancing motions were seldom explored in previous studies. So we explore its solution in this work.

3 Multi-modal Collaborative Learning Encoder from Motions and Music

In this work, we propose a multi-modal collaborative learning encoder method (MCLE) to enhance music recommendation performance. MCLE simultaneously optimizes feature extraction from both dancing motion data and its background music data using collaborative learning. Fig. 1 illustrates architecture of the proposed MCLEMCD method. The method is composed of two parts: a motion encoder and a music encoder. Graph convolutional network based motion encoder takes dancing motion sequences as inputs, extracts deep features, and performs action classification after applying a full connection layer. Music auto-encoder takes MFCCs features as inputs, and extracts deep music features. Two parts are co-optimized by mapping to same embedded feature space. By aligning two modalities, both motion and music features are able to reconstruct MFCCs features, which are further exploited for music classification.

3.1 Motion Encoder

Motion encoder extracts motion features using a graph convolutional network based model [42]. Extracted motion features by motion encoder are further processed through fully connected layers for motion classification. Model parameters are optimized using motion classification. Loss for motion classification task is defined as a cross-entropy loss l_{motion}:

$$l_{motion} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{M} y_{ic} log(p_{ic}), \tag{1}$$

where N denotes number of samples, M stands for number of motion classes, y_{ic} denotes whether predicted motion label is the same as ground truth label (y_{ic} is 1 if they are the same, and 0 otherwise), and p_{ic} is probability of predicted label y_{ic}.

In addition to key joint coordinates of human skeleton, second-order information (such as body limb length and direction) representing limb information is also informative and beneficial for action recognition tasks. Therefore, we employ limb vectors to denote limb lengths and directions. In section 4, we evaluate performance of four feature configurations: joint positions, second-order limb feature, early fusion of two features, and later fusion of two features. Experimental results show that late fusion of joint and limb features achieves the best performance for MusicToDance dataset, and joint positions give the best performance for AIST++ dataset.

After 1500 epoches, motion classification loss and prediction accuracy reach plateau. Motion classification accuracy achieves 57.06% on test set of AIST++ dataset, and 93.96% on test set of MusicToDance dataset. Feature extraction of motion encoder will be further optimized in the following co-optimization step.

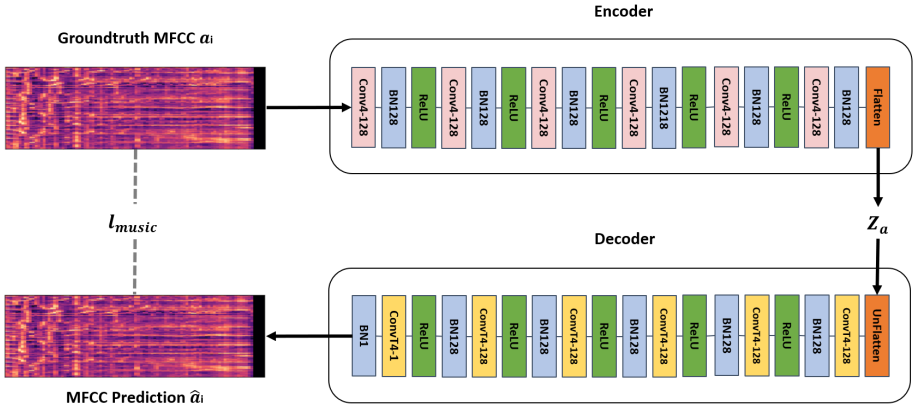

Fig. 2 Architecture of music auto-encoder in the proposed MCLEMCD model. Encoder part consists of 2D convolutions, BatchNorm operations, and ReLU activations, while decoder part consists of 2D deconvolutions, BatchNorm operations, and ReLu activations.

3.2 Music Auto-encoder

After extracting MFCCs features from original audio waveforms, we adopt a COALA [39] based music auto-encoder to prepare for motion-music alignment procedure. Encoder consists of 2D convolutional layers, BatchNorm layers, and ReLU layers. After encoder, features are reshaped as a 2D feature. Decoder, on the other hand, applies reverse operations of encoder. It also reshapes feature dimensions, and then carries out deconvolutions, BathNorm operations, and ReLU activations.

Flow chart of music auto-encoder is shown in Fig. 2. Music encoder extracts feature Z_a with decreased dimensions. In decoder, extracted feature Z_a is transformed back to its original dimensions. Goal of auto-encoder is that reconstructed feature is similar to original input feature. Loss of music auto-encoder is defined as a mean squared error between input and its reconstruction:

$$l_{music} = \frac{1}{N}\sum_{i=1}^{N}(\hat{a}_i - a_i)^2, \tag{2}$$

where $\hat{a}_i$ denotes reconstructed feature through music auto-encoder, and a_i denotes input feature.

We carry out ablation studies to explore music auto-encoder architecture. We set up two additional models: one is additional Dropout layers on top of music autoencoder to prevent model from overfitting; the other is a fully connected network between motion encoder and music encoder. Experimental results show that additional dropout layer reduce reconstruction accuracy, and the accuracy fluctuates to a certain extent. While additional fully connected layer helps improve accuracy.

3.3 Multi-modal Co-optimization

Collaborative learning from motion and music data is carried out through a co-optimization procedure within deep learning framework. Motion encoder and music encoder are co-optimized with a mean squared loss between extracted motion and music features:

$$l_{pair} = \frac{1}{N}\sum_{i=1}^{N}(Z_{im} - Z_{ia})^2, \tag{3}$$

where Z_{im} denotes motion feature of the i-th sample extracted using motion encoder, and Z_{ia} denotes music feature of the i-th sample extracted through music encoder.

In addition to pair loss l_{pair}, we further reconstruct music feature from motion feature, and incorporate reconstruction loss l_{mo2mu} in co-optimization:

$$l_{mo2mu} = \frac{1}{N}\sum_{i=1}^{N}(\hat{m}_i - a_i)^2, \tag{4}$$

where $\hat{m}_i$ denotes reconstructed music feature from motion feature Z_{im}, and a_i denotes input music feature.

During training, motion and music encoders are optimized separately. After their losses reach plateaus, respectively, features of these two media are co-optimized. The independent training and collaborative learning are illustrated in Fig. 1. Independent motion and music training processes follow data flow marked with black lines, while co-optimization process is illustrated as a data flow marked with red lines. After co-optimization process, motion and music data are mapped to embedded representations (i.e., Z_m or Z_a) with the same dimensions. Experimental results show that motion classification accuracy is improved with collaborative learning module.

3.4 MCLEMCD applied to Music Recommendation

We analyse dancing motions and recommend their accompanying music using predicted motion/music categories. The proposed MCLEMCD model improves music recommendation performance by collaborative learning from motion and music modalities. Through above-mentioned co-optimization procedure, we incorporate music cues in motion features, so that motion/music classification performance is improved. Thus, music recommendation performance is enhanced accordingly.

4 Experiments

In this section, we carry out qualitative and quantitative evaluations of the proposed MCLEMCD method. There are relatively few publicly available datasets

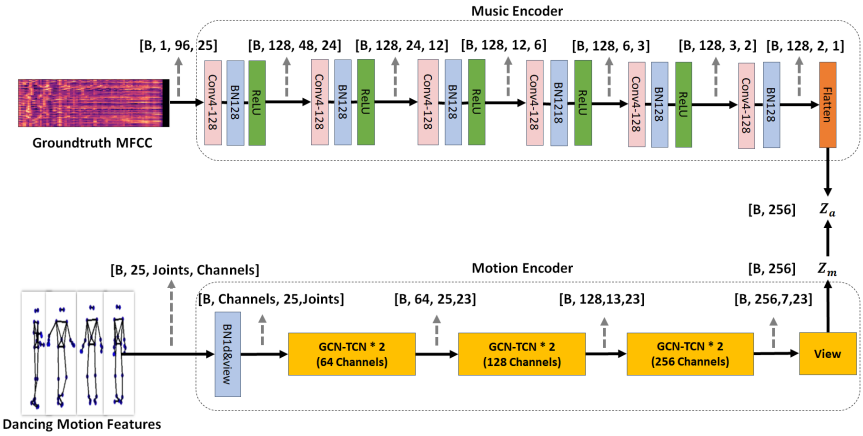

Fig. 3 Dimensional Variations of Alignment Network for Music Encoder and Motion Encoder. "B" denotes number of batches, and "channels" in GCN-TCN module of motion encoder indicate number of convolutional channels.

consisting of 3D dancing motions and their accompanying music audio files. In this work, we use two popular datasets for evaluation.

We experiment with two variants of the proposed model illustrated in Fig. 1. A dropout layer is added after each ReLU activation function in music encoder. We also experiment with a variant by adding fully connected layers as an indirect mapping network between output of motion encoder Z_m and output of music encoder Z_a. This fully connected network takes deep motion feature Z_m as input, and maps feature channels from 256 to 512, 1024, 2048, and 4096, respectively. Then another stack of fully connected layers apply reverse operations and maps feature channels back to 256, which is feature dimension of Z_a.

4.1 Parameter Settings for MusicToDance Dataset

MusicToDance Dataset [43], collected by Tsinghua University, consists of 60 motion-music pairs in total, or 907, 200 frames, including four dancing types (i.e., waltz, tango, cha-cha, and rumba). Frame rate of motion data is 25 fps. Sampling rate of music data is $44.1K$ Hz. In experiments, we set sampling rate to $12.8K$ Hz so that length of extracted music feature match with that of extracted motion feature. Human poses are represented using 25 joint positions, and we take 23 of them.

The dataset is unbalanced and collected videos of the dataset are of various lengths. The minimum length of collected videos is 229 frames, so we first split videos into segments of 229 frames, and select videos from each category so that the dataset has balanced classes and the model processes unified input data. After segmentation, each category contains 34 segments of

motion-music pairs. MusicToDance dataset does not contain standard training/validation/test splits. We split the dataset evenly into three subsets: 45 pairs for training, 45 for validation, and 46 for test.

Music auto-encoder is mainly composed of 2D convolutional layers, BatchNorm layers, and ReLU activation layers. All layers retain data dimensions except convolutional layers. Dimension variation process is shown in Fig. 3. We denote input dimensions of music encoder as $[N \times C \times N_{MFCCs} \times T]$, where N represents batch size and is set as 72, C is number of channels and is set as 1, N_{MFCCs} stands for length of MFCCs features, and T denotes length of music sequence. We employ Librosa [44] music processing library to extract MFCCs feature and set feature dimension as 96, so N_{MFCCs} is equal to 96. Collaborative learning requires that the dancing motion feature Z_m and the music feature Z_a must be of the same dimensions, so the input frame length of the music encoder is set as 25. Note that, without incorporating collaborative learning, motion/music sequences are divided into fragments of 229 frames; with collaborative learning, motion/music sequences are divided into segments of 25 frames. The dataset is re-split for collaborative learning and includes 405 pairs for training, 405 for validation, and 414 for test.

For motion data, its input feature dimensions are $[N \times T \times N_j \times N_d]$, where N_j denotes number of body joints and is equal to 23, N_d stands for human motion feature in a frame, and N_d is equal to 3 for human body joints/limbs and is equal to 6 for combined feature of human body joints and limbs. Motion categories use one-hot encoding. For both motion and music data, input features are normalized, and is scaled in range of $[0 - 1]$.

The optimum epoch numbers are 3000. Motion encoder and music auto-encoder are first trained separately for the first 1500 epochs. Co-optimization procedure is run for the following 1500-3000 epochs. We use an Adam optimizer and learning rate is set as $5e - 4$.

4.2 Parameter Settings for AIST++ Dataset

AIST++ dance motion dataset [45] consists of $1,408$ 3D human dancing sequences, described by root joint trajectories and joint angle rotations. It contains 10 dancing types, including old school (break, pop, lock, and waack) and new school (middle hip-hop, LA-style hip-hop, house, krump, street jazz, and ballet jazz). Dancing motion videos last from 7.4 seconds to 48 seconds. Frame rate of motion data is 60 fps. Sample rate of music inputs is $48K$Hz. In experiments, we set sampling rate to $30.7K$ Hz so that length of extracted music feature match with that of extracted motion feature. Human poses are represented using 17 joint positions.

The dataset provides standard training/validation/test splits, including 980 training samples, 20 validation samples, and 20 test samples. Following the same protocol, we first divide motion and music videos into segments of 425 frames. Then, we further split music segments into a unit of 25 frames. Finally, the dataset is divided into three splits, with 16660 pairs for training, 340 for validation, and 340 for test.

4.3 Experiment Design

We design four experiments with various motion feature configurations for MusicToDance dataset. Table 1 lists detailed configurations of experiment 1-4.

Table 1 Settings of Experiment 1-4 on MusicToDance Dataset.

Experiment ID	Input Dimensions	Feature and Model Settings
1	$414 \times 3 \times 25 \times 23$	Motion Feature (joints)
2		Motion Feature (limbs)
3		Motion Model Fusion (Exp1 & Exp2)
4	$414 \times 6 \times 25 \times 23$	Motion Feature (joints & limbs)

- Experiment 1: Input dimensions of human body joint positions are $[N \times N_d \times T \times N_j]$, where N denotes number of samples and is equal to 414 for test, N_d stands for number of dimensions and is equal to 3, T is number of frames of a video segment and is equal to 25, and N_j represents number of body joints and is equal to 23. Input music feature dimensions are $[N \times 1 \times N_{MFCCs} \times T]$, where N_{MFCCs} is equal to 96 and stands for dimension of extracted MFCCs feature. The two features are fed into motion encoder and music auto-encoder for collaborative learning.
- Experiment 2: In this experiment, motion feature uses human body limb positions. Other settings are the same as Experiment 1.
- Experiment 3: Predictions from Experiment 1 and Experiment 2 are fused to form a new prediction.
- Experiment 4: Human body joints and limb positions are concatenated to denote dancing motions. Final motion feature dimensions are $[N \times N_d \times T \times N_j]$, where N_d is equal to 6 (human body joint feature takes 3 and human body limb feature takes 3). Other settings are the same as Experiment 1.

We design four other experiments (Experiments 5-8) for AIST++ dataset. In experiment 5-8, N is equal to 16660 for training set of AIST++ dataset and 340 for test set, and N_j is equal to 17. Other parameter settings are the same as those of MusicToDance dataset. Table 2 provides specific configurations of experiment 5-8.

Table 2 Settings of Experiment 5-8 on AIST++ Dataset.

Experiment ID	Input Dimensions	Feature and Model Settings
5	$340 \times 3 \times 25 \times 17$	Motion Feature (joints)
6		Motion Feature (limbs)
7		Motion Model Fusion (Exp1 & Exp2)
8	$340 \times 6 \times 25 \times 17$	Motion Feature (joints & limbs)

Table 3 Performances of Experiment 1-4 on MusicToDance Dataset (CL means collaborative learning).

With/Without CL	Method	Experiment ID			
		Experiment 1	Experiment 2	Experiment 3	Experiment 4
Without CL	MCLEMCD	77.54%	92.27%	93.96%	87.68%
	MCLEMCD-Dropout	88.16%	93.48%	95.17%	92.27%
With CL	MCLEMCD	90.58%	96.62%	96.68%	93.72%
	MCLEMCD-Dropout	89.86%	95.41%	96.38%	97.10%
	MCLEMCD-FC	88.89%	97.83%	98.07%	93.48%

Table 4 Performances of Experiment 5-8 on AIST++ Dataset (CL means collaborative learning).

With/Without CL	Method	Experiment ID			
		Experiment 5	Experiment 6	Experiment 7	Experiment 8
Without CL	MCLEMCD	57.94%	50.88%	57.06%	48.53%
	MCLEMCD-Dropout	56.76%	15.88%	49.71%	52.65%
With CL	MCLEMCD	60.29%	57.35%	63.82%	53.24%
	MCLEMCD-Dropout	59.71%	17.06%	52.35%	55.29%
	MCLEMCD-FC	63.53%	56.18%	65.29%	50.00%

Several different models are evaluated on each of these experiments. Evaluated models include variants with and without collaborative learning to verify their effectiveness.

4.4 Quantitative Evaluation

We calculate ratio of correctly predicted music categories over all test samples as prediction accuracy. Table 3 and table 4 list performance of Experiment 1-4 on MusicToDance dataset and Experiment 5-8 on AIST++ dataset, respectively. From the table, we observe that collaborative learning enhances performances dramatically. For example, music recommendation without collaborative learning (CL) [42] achieves the highest prediction accuracy (93.96%) in experiment 3, while the proposed MCLEMCD-FC method achieves 98.07% and is 4.11% higher than method without CL. For evaluations on AIST++ dataset, the highest prediction accuracy (65.29%) is also achieved by model with co-optimization and additional fully connected layers.

Overall performance of MusicToDance dataset is much better than that of AIST++ datset, because AIST++ dataset consists of more categories. Furthermore, most of the dance music in AIST++ dataset are relatively fast-paced, and music pieces from different categories are very similar so it is difficult to distinguish among categories.

We also display confusion matrices in Fig. 4 and Fig. 5 to further demonstrate performance listed in table 3 and table 4. Fig. 4 plots confusion matrix of experiment with the highest accuracy in table 3, i.e., experiment 3 using

MCLEMCD-FC method with CL. We observe that the proposed method performs well for all categories. While, some categories achieve higher accuracy than others, e.g., "cha-cha" outperforms "rumba". The proposed MCLEMCD method outperforms music recommendation method without CL with a margin of 10% on "waltz".

Fig. 5 plots confusion matrix for experiment with the highest accuracy in table 4, i.e., experiment 3 using MCLEMCD-FC method with CL. We observe that predictions of "hip hop" music and "jazz" music are more accurate, while predictions of other music genres, such as "break", "krump" and "house", are less accurate.

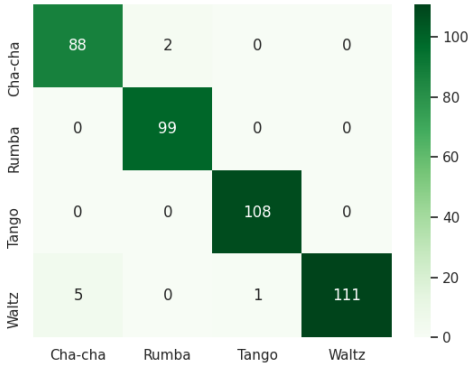

Fig. 4 Confusion Matrix of MusicToDance Dataset.

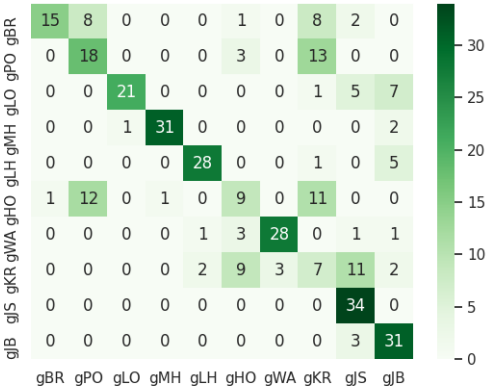

Fig. 5 Confusion Matrix of AIST++ Dataset.

Table 5 Quantitative Performance of MusicToDance Dataset with the Same Settings as the Qualitative Evaluation (CL means collaborative learning).

Experiment ID	Prediction Accuracy without CL		Prediction Accuracy with CL		
	MCLEMCD	MCLEMCD-Dropout	MCLEMCD	MCLEMCD-Dropout	MCLEMCD-FC
1	84.78%	95.65%	95.65%	93.48%	95.65%
2	95.65%	95.65%	97.82%	100.00%	97.82%
3	95.65%	95.65%	100.00%	100.00%	100.00%
4	93.48%	95.65%	97.82%	100.00%	95.65%

Table 6 Quantitative Performance of AIST++ Dataset with the Same Settings as Qualitative Evaluation (CL means collaborative learning).

Experiment ID	Prediction Accuracy without CL		Prediction Accuracy with CL		
	MCLEMCD	MCLEMCD-Dropout	MCLEMCD	MCLEMCD-Dropout	MCLEMCD-FC
5	70.00%	65.00%	80.00%	85.00%	80.00%
6	70.00%	55.00%	70.00%	60.00%	65.00%
7	80.00%	65.00%	85.00%	70.00%	85.00%
8	65.00%	70.00%	60.00%	70.00%	70.00%

Before qualitative evaluation, we re-evaluate quantitative performance so that qualitative evaluations are under the same settings. Performances provided in table 3 and table 4 are evaluated per segment. Table 5 and table 6 list performances per music/dancing-motion piece. We count recommended categories after each test segment in test music/dancing-motion piece, and select category with the highest occurrences as recommended category for a whole piece. As shown in table 5 and table 6, prediction accuracies per music/dancing-motion piece with CL are more accurate than without, and prediction accuracies per music/dancing-motion piece are higher than prediction accuracies per segment in table 3 and table 4. This indicates that by fusing recommendations from each segment, overall recommendation accuracy is further enhanced.

4.5 Qualitative Evaluation

For qualitative evaluation, we employ MCLEMCD-FC method with joint and limb features as inputs. The method recommends three candidate music pieces for each test dancing motion. We invite nine subjects to rate performance of recommendations online. These subjects are from music (3 subjects), dance (3 subjects), and computer science (3 subjects) majors, corresponding to subject ID 1-9 in qualitative evaluation table. Dancing videos accompanied by ground truth music pieces are first displayed. Then dancing videos with recommended background music are displayed. Subjects are asked to rate recommendation performance based on following questions.

- Q1: Whether recommended background music is consistent with style of dancing motions? Scoring ranges from 1 to 5, where 1 denotes that dancing motions and music are not consistent in style at all, 5 denotes that style of dancing motions and recommended background music completely match

with each other, and 2, 3 and 4 represent increasing consistencies in style between 1 and 5.
- Q2: Whether recommended background music matches tempo of dancing motions? Rating is also on a 5-point scale. Rating standard is the same as that of Q1.
- Q3: Whether recommended music is more suitable for dancing motions than ground truth music? Rating is either 1 (recommended music is more suitable for dancing motions than ground truth music) or 0 (otherwise).

Based on ratings from all subjects, we calculate average score of each category by each subject and get "per subject" score. Sum of "per subject" values over a major are then averaged to obtain "per major" score for each category. Finally, "per major" scores are added and averaged to calculate "average" value of a category.

Qualitative evaluation results are shown in table 7 and table 8. Based on average value of Q3 in table 7 and table 8, we observe that recommended music is considered as being more suitable for dancing motions than ground truth background music.

We observe from table 7 that both "rumba" and "tango" score higher in style and tempo consistency with CL, and subjects consider that recommended music to be more suitable for dancing motions compared with recommended music without CL [42]. But for "cha-cha" and "waltz", ratings are not as high as their counterparts. We argue that this difference is because subjects tend to be more critical during online assessment. In addition, subjects from music and dance majors rate lower in style and tempo than subjects from computer science majors. Overall, qualitative evaluation proves effectiveness of the proposed method because most subjects from music and dance majors believe that recommended music is more suitable for dancing motions than ground truth music, even though music style and tempo do not completely match dancing motions.

From table 8, we observe that subjects rate higher scores for "hip-hop" category, and they believe that recommended music is more suitable for dancing motions in both style and tempo for this category. While "house" and "break" categories are rated the lowest scores for Q3. Ratings are consistent with confusion matrix of qualitative evaluation result in Fig. 5. From the table, we also observe that many scores are around 0.5, showing that AIST++ dataset is difficult to distinguish among music categories. This is also consistent with the low prediction accuracy of AIST++ dataset compared with the high prediction accuracy of MusicToDance dataset.

5 Discussions and Future Works

This work proposes a MCLEMCD method that improves prediction accuracy of music recommendation through co-training of two modalities, i.e., dancing motions and music. In addition, we evaluate the proposed model and its variants by adding dropout layers and a fully connected mapping network.

Table 7 Qualitative Evaluation Results of MusicToDance Dataset.

Category	Subject ID	Style Quality (Q1)			Tempo Quality (Q2)			Quality Compared with GT (Q3)		
		Per Subject	Per Major	Average	Per Subject	Per Major	Average	Per Subject	Per Major	Average
Cha-cha	1	0.62	0.71	0.76	0.60	0.72	0.77	+0.67	+0.63	-0.47
	2	0.78			0.76			+0.67		
	3	0.73			0.80			+0.56		
	4	0.76	0.74		0.76	0.78		+0.56	+0.56	
	5	0.76			0.78			+0.56		
	6	0.71			0.80			+0.56		
	7	0.73	0.83		0.64	0.81		-0.22	-0.22	
	8	0.96			0.96			-0.22		
	9	0.80			0.84			~~+0.67~~[1]		
Rumba	1	0.46	0.69	0.78	0.80	0.79	0.77	+0.67	+0.74	+0.67
	2	0.80			0.87			+0.78		
	3	0.80			0.69			+0.78		
	4	0.76	0.81		0.69	0.70		+0.89	+0.82	
	5	0.86			0.71			+0.78		
	6	0.82			0.71			+0.78		
	7	0.73	0.85		0.73	0.83		-0.22	-0.44	
	8	0.98			0.96			-0.44		
	9	0.84			0.82			+0.67		
Tango	1	0.50	0.69	0.76	0.77	0.80	0.81	+0.67	+0.67	+0.67
	2	0.73			0.83			+0.67		
	3	0.83			0.80			+0.67		
	4	0.83	0.79		0.73	0.79		+0.67	+0.67	
	5	0.80			0.80			+0.67		
	6	0.73			0.83			+0.67		
	7	0.70	0.81		0.67	0.83		~~-0.16~~[1]	+0.67	
	8	0.90			0.93			±0.50		
	9	0.83			0.90			0.83		
Waltz	1	0.62	0.76	0.82	0.78	0.73	0.72	+0.67	+0.56	+0.61
	2	0.78			0.82			-0.33		
	3	0.88			0.60			+0.67		
	4	0.87	0.88		0.64	0.61		+0.78	+0.59	
	5	0.89			0.58			-0.44		
	6	0.87			0.60			+0.56		
	7	0.73	0.81		0.67	0.83		~~-0.00~~[1]	+0.67	
	8	0.91			1.00			+0.67		
	9	0.80			0.82			+0.67		

[1] This is considered as an outlier (0.4 difference from average). Noisy data are not taken into consideration for calculating "per major" and "average" values.

Experimental results show that dropout module does not significantly improve accuracy, while adding a fully connected mapping network between music and motion encoders can significantly improve prediction accuracy. The proposed method is validated on MusicToDance and AIST++ datasets using quantitative and qualitative evaluation metrics. Experimental results show that the

Table 8 Qualitative Evaluation Results of AIST++ Dataset (Split into Two Parts) - Part 1

Category	Subject ID	Style Quality (Q1)			Tempo Quality (Q2)			Quality Compared with GT (Q3)		
		Per Subject	Per Major	Average	Per Subject	Per Major	Average	Per Subject	Per Major	Average
BR	1	0.67	0.69	0.75	0.57	0.65	0.69	+0.67	+0.56	-0.48
	2	0.80			0.77			±0.50		
	3	0.60			0.60			±0.50		
	4	0.57	0.67		0.60	0.60		±0.50	+0.56	
	5	0.73			0.60			±0.50		
	6	0.73			0.60			+0.67		
	7	0.83	0.90		0.77	0.81		-0.17	-0.33	
	8	1.00			0.87			-0.33		
	9	0.87			0.80			±0.50		
HO	1	0.70	0.76	0.78	0.73	0.73	0.75	-0.17	-0.17	-0.43
	2	0.80			0.77			-0.17		
	3	0.77			0.70			~~+0.83~~[1]		
	4	0.70	0.70		0.70	0.70		+0.83	+0.72	
	5	0.70			0.70			+0.67		
	6	0.70			0.70			+0.67		
	7	0.80	0.87		0.80	0.83		±0.50	-0.39	
	8	0.97			0.87			-0.17		
	9	0.83			0.83			±0.50		
JB	1	0.63	0.63	0.63	0.63	0.69	0.70	±0.50	+0.56	-0.38
	2	0.67			0.80			+0.67		
	3	0.60			0.63			±0.50		
	4	0.60	0.60		0.63	0.63		±0.50	±0.50	
	5	0.60			0.63			±0.50		
	6	0.60			0.63			±0.50		
	7	0.53	0.65		0.57	0.79		-0.00	-0.09	
	8	0.63			0.93			-0.17		
	9	0.80			0.87			~~+0.67~~[1]		
JS	1	0.63	0.65	0.67	0.57	0.67	0.71	±0.50	+0.56	+0.68
	2	0.83			0.83			±0.50		
	3	0.50			0.60			+0.67		
	4	0.50	0.51		0.60	0.60		+0.83	+0.72	
	5	0.50			0.60			+0.83		
	6	0.53			0.60			±0.50		
	7	0.70	0.86		0.77	0.86		~~-0.33~~[1]	+0.75	
	8	1.00			0.87			+0.67		
	9	0.87			0.93			+0.83		
KR	1	0.70	0.75	0.79	0.80	0.82	0.78	+0.83	+0.92	+0.68
	2	0.83			0.87			~~±0.50~~[1]		
	3	0.73			0.80			+1.00		
	4	0.73	0.72		0.80	0.80		+1.00	+0.94	
	5	0.73			0.80			+0.83		
	6	0.70			0.80			+1.00		
	7	0.73	0.89		0.77	0.73		-0.33	-0.17	
	8	0.97			0.50			-0.00		
	9	0.97			0.93			~~+0.67~~[1]		

[1] This is considered as an outlier (0.4 difference from average). Noisy data are not taken into consideration for calculating "per major" and "average" values.

Qualitative Evaluation Results of AIST++ Dataset (Split into Two Parts) - Part 2

Category	Subject ID	Style Quality (Q1)			Tempo Quality (Q2)			Quality Compared with GT (Q3)		
		Per Subject	Per Major	Average	Per Subject	Per Major	Average	Per Subject	Per Major	Average
LH	1	0.67	0.73	0.76	0.67	0.71	0.75	+0.83	+0.75	+0.70
	2	0.80			0.80			~~-0.33~~[1]		
	3	0.73			0.67			+0.67		
	4	0.67	0.68		0.67	0.67		+0.67	+0.78	
	5	0.67			0.67			+0.83		
	6	0.70			0.67			+0.83		
	7	0.80	0.88		0.83	0.86		±0.50	+0.56	
	8	1.00			0.86			±0.50		
	9	0.83			0.90			+0.67		
LO	1	0.80	0.79	0.80	0.73	0.73	0.77	+0.67	+0.67	+0.70
	2	0.90			0.77			+0.67		
	3	0.67			0.70			+0.67		
	4	0.70	0.68		0.70	0.72		+0.67	+0.67	
	5	0.67			0.73			+0.67		
	6	0.67			0.73			+0.67		
	7	0.83	0.92		0.77	0.85		~~-0.17~~[1]	+0.75	
	8	1.00			0.90			+0.67		
	9	0.93			0.87			+0.83		
MH	1	0.77	0.73	0.76	0.70	0.75	0.76	+1.00	+0.92	+0.86
	2	0.73			0.87			~~-0.33~~[1]		
	3	0.70			0.67			+0.83		
	4	0.70	0.70		0.67	0.64		+0.83	+0.83	
	5	0.70			0.63			+0.83		
	6	0.70			0.63			+0.83		
	7	0.80	0.86		0.70	0.89		~~-0.17~~[1]	+0.83	
	8	0.97			1.00			+0.83		
	9	0.80			0.97			+0.83		
PO	1	0.67	0.73	0.74	0.80	0.79	0.78	+0.83	+0.72	+0.66
	2	0.90			0.87			±0.50		
	3	0.63			0.70			+0.83		
	4	0.63	0.63		0.70	0.70		+0.83	+0.83	
	5	0.63			0.70			+0.83		
	6	0.63			0.70			+0.83		
	7	0.77	0.87		0.72	0.84		-0.33	-0.44	
	8	1.00			0.93			±0.50		
	9	0.83			0.87			±0.50		
WA	1	0.57	0.65	0.69	0.53	0.62	0.62	-0.33	+0.55	-0.42
	2	0.77			0.87			±0.50		
	3	0.60			0.47			+0.83		
	4	0.60	0.60		0.47	0.47		+0.83	+0.72	
	5	0.60			0.47			+0.83		
	6	0.60			0.47			±0.50		
	7	0.70	0.81		0.67	0.78		-0.00	-0.00	
	8	0.87			0.77			-0.00		
	9	0.87			0.90			~~+0.67~~[1]		

[1] This is considered as an outlier (0.4 difference from average). Noisy data are not taken into consideration for calculating "per major" and "average" values.

model performs well on MusicToDance dataset, while results on AIST++ are moderate. The reason may be that AIST++ dataset contains more categories, and most categories are similar in style and tempo (all belong to popular rhythm music). As a result, the dataset is difficult to distinguish among categories.

Our further research will be conducted in two directions. One is to continue to explore network structure to improve its performance on AIST++ dataset. The other is to establish a mapping between dancing motion feature and music feature, and directly generate music from dancing motion feature.

Acknowledgments

Jordi Gonzàlez acknowledges the support by the Spanish Ministry of Economy and Competitiveness (MINECO) and the European Regional Development Fund (ERDF) under Project PID2020-120311RB-I00/AEI/10.13039/501100011033.

References

[1] Goto, M., Dannenberg, R.B.: Music interfaces based on automatic music signal analysis: New ways to create and listen to music. IEEE Signal Processing Magazine **36**, 74–81 (2019)

[2] Schedl, M.: Intelligent user interfaces for social music discovery and exploration of large-scale music repositories. Proceedings of the 2017 ACM Workshop on Theory-Informed User Modeling for Tailoring and Personalizing Interfaces (2017)

[3] Oramas, S., Nieto, O., Barbieri, F., Serra, X.: Multi-label music genre classification from audio, text and images using deep features. In: ISMIR (2017)

[4] Mayer, R., Rauber, A.: Music genre classification by ensembles of audio and lyrics features. In: ISMIR (2011)

[5] Cai, X., Zhang, H.: Music genre classification based on auditory image, spectral and acoustic features. Multimedia Systems, 1–13 (2022)

[6] Chaturvedi, V., Kaur, A.B., Varshney, V., Garg, A., Chhabra, G.S., Kumar, M.: Music mood and human emotion recognition based on physiological signals: a systematic review. Multimedia Systems, 1–24 (2021)

[7] Yang, Y.-H., Chen, H.H.: Machine recognition of music emotion: A review. ACM Trans. Intell. Syst. Technol. **3**, 40–14030 (2012)

[8] Huq, A., Bello, J.P., Rowe, R.: Automated music emotion recognition: A systematic evaluation. Journal of New Music Research **39**, 227–244 (2010)

[9] Knees, P., Schedl, M.: Music similarity and retrieval: An introduction to audio- and web-based strategies. (2016)

[10] Karydis, I., Kermanidis, K.L., Sioutas, S., Iliadis, L.S.: Comparing content and context based similarity for musical data. Neurocomputing **107**, 69–76 (2013)

[11] Krumhansl, C.L., Schenck, D.L.: Can dance reflect the structural and expressive qualities of music? a perceptual experiment on balanchine's choreography of mozart's divertimento no. 15. Musicae Scientiae **1**(1), 63–85 (1997). https://doi.org/10.1177/102986499700100105

[12] Su, Y.-H.: Rhythm of music seen through dance: Probing music–dance coupling by audiovisual meter perception (2017). https://doi.org/10.31234/osf.io/ujkq9

[13] Alemi, O., Françoise, J., Pasquier, P.: Groovenet: Real-time music-driven dance movement generation using artificial neural networks. (2017)

[14] Lee, J., Kim, S., Lee, K.: Listen to dance: Music-driven choreography generation using autoregressive encoder-decoder network. CoRR **abs/1811.00818** (2018) https://arxiv.org/abs/1811.00818

[15] Manfré, A., Infantino, I., Vella, F., Gaglio, S.: An automatic system for humanoid dance creation. In: BICA 2016 (2016)

[16] Fan, R., Xu, S., Geng, W.: Example-based automatic music-driven conventional dance motion synthesis. IEEE Transactions on Visualization and Computer Graphics **18**(3), 501–515 (2012). https://doi.org/10.1109/TVCG.2011.73

[17] Lee, M., Lee, K., Park, J.: Music similarity-based approach to generating dance motion sequence. Multimedia Tools and Applications **62**, 895–912 (2012)

[18] Ofli, F., Erzin, E., Yemez, Y., Tekalp, A.M.: Learn2dance: Learning statistical music-to-dance mappings for choreography synthesis. IEEE Transactions on Multimedia **14**(3), 747–759 (2012). https://doi.org/10.1109/TMM.2011.2181492

[19] Lee, H.-Y., Yang, X., Liu, M.-Y., Wang, T.-C., Lu, Y.-D., Yang, M.-H., Kautz, J.: Dancing to music. In: Wallach, H., Larochelle, H., Beygelzimer, A., d' Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 32. Curran Associates, Inc., ??? (2019). https://proceedings.neurips.cc/paper/2019/file/7ca57a9f85a19a6e4b9a248c1daca185-Paper.pdf

[20] Tsuchida, S., Fukayama, S., Goto, M.: Query-by-Dancing: A Dance Music Retrieval System Based on Body-Motion Similarity, pp. 251–263 (2019). https://doi.org/10.1007/978-3-030-05710-7_21

[21] Ohkushi, H., Ogawa, T., Haseyama, M.: Music recommendation according to human motion based on kernel cca-based relationship. EURASIP Journal on Advances in Signal Processing **2011** (2011). https://doi.org/10.1186/1687-6180-2011-121

[22] Davis, S., Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Transactions on Acoustics, Speech, and Signal Processing **28**(4), 357–366 (1980). https://doi.org/10.1109/TASSP.1980.1163420

[23] Schörkhuber, C.: Constant-q transform toolbox for music processing. (2010)

[24] Maximum Filter Vibrato Suppression for Onset Detection. In: Proceedings of the 16th International Conference on Digital Audio Effects (DAFx-13), Maynooth, Ireland (2013)

[25] Grosche, P., Müller, M., Kurth, F.: Cyclic tempogram—a mid-level tempo representation for musicsignals. In: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 5522–5525 (2010). https://doi.org/10.1109/ICASSP.2010.5495219

[26] Bae, H.-S., Lee, H.-J., Lee, S.-G.: Voice recognition based on adaptive mfcc and deep learning. In: 2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA), pp. 1542–1546 (2016). IEEE

[27] Deng, M., Meng, T., Cao, J., Wang, S., Zhang, J., Fan, H.: Heart sound classification based on improved mfcc features and convolutional recurrent neural networks. Neural Networks **130**, 22–32 (2020)

[28] Boles, A., Rad, P.: Voice biometrics: Deep learning-based voiceprint authentication system. In: 2017 12th System of Systems Engineering Conference (SoSE), pp. 1–6 (2017). IEEE

[29] Shiratori, T., Nakazawa, A., Ikeuchi, K.: Synthesizing dance performance using musical and motion features. In: Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006., pp. 3654–3659 (2006). https://doi.org/10.1109/ROBOT.2006.1642260

[30] Su, Y.-H.: Rhythm of music seen through dance: Probing music–dance coupling by audiovisual meter perception (2017). https://doi.org/10.31234/osf.io/ujkq9

[31] Verma, P., Sah, A., Srivastava, R.: Deep learning-based multi-modal approach using rgb and skeleton sequences for human activity recognition. Multimedia Systems **26**(6), 671–685 (2020)

[32] Chu, W.-T., Tsai, S.-Y.: Rhythm of motion extraction and rhythm-based cross-media alignment for dance videos. IEEE Transactions on Multimedia **14**(1), 129–141 (2012). https://doi.org/10.1109/TMM.2011.2172401

[33] Rubinstein, M.: Analysis and visualization of temporal variations in video. PhD thesis, Massachusetts Institute of Technology (Feb 2014)

[34] Shi, L., Zhang, Y., Cheng, J., Lu, H.: Adaptive spectral graph convolutional networks for skeleton-based action recognition. CoRR **abs/1805.07694** (2018) https://arxiv.org/abs/1805.07694

[35] Sarkar, A.: Applying co-training methods to statistical parsing. In: Second Meeting of the North American Chapter of the Association for Computational Linguistics (2001). https://aclanthology.org/N01-1023

[36] Levin, Viola, Freund: Unsupervised improvement of visual detectors using cotraining. In: Proceedings Ninth IEEE International Conference on Computer Vision, pp. 626–6331 (2003). https://doi.org/10.1109/ICCV.2003.1238406

[37] Christoudias, C.M., Saenko, K., Morency, L.-P., Darrell, T.: Co-adaptation of audio-visual speech and gesture classifiers. In: ICMI '06 (2006)

[38] Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. Proceedings of the Annual ACM Conference on Computational Learning Theory (2000). https://doi.org/10.1145/279943.279962

[39] Favory, X., Drossos, K., Virtanen, T., Serra, X.: COALA: co-aligned autoencoders for learning semantically enriched audio representations. CoRR **abs/2006.08386** (2020) https://arxiv.org/abs/2006.08386

[40] Lee, H.-Y., Yang, X., Liu, M.-Y., Wang, T.-C., Lu, Y.-D., Yang, M.-H., Kautz, J.: Dancing to Music (2019)

[41] Qi, Y., Liu, Y., Sun, Q.: Music-driven dance generation. IEEE Access **7**, 166540–166550 (2019). https://doi.org/10.1109/ACCESS.2019.2953698

[42] Gong, W., Yu, Q.: A deep music recommendation method based on human motion analysis. IEEE Access **9**, 26290–26300 (2021). https://doi.org/10.1109/ACCESS.2021.3057486

[43] Tang, T., Jia, J., Hanyang, M.: Dance with melody: An lstm-autoencoder approach to music-oriented dance synthesis, pp. 1598–1606 (2018). https://doi.org/10.1145/3240508.3240526

[44] McFee, B., Metsai, A., McVicar, M., Balke, S., Thomé, C., Raffel, C., Zalkow, F., Malek, A., Dana, Lee, K., Nieto, O., Ellis, D., Mason, J., Battenberg, E., Seyfarth, S., Yamamoto, R., viktorandreevichmorozov, Choi, K., Moore, J., Bittner, R., Hidaka, S., Wei, Z., nullmightybofo, Weiss, A., Hereñú, D., Stöter, F.-R., Friesch, P., Vollrath, M., Kim, T., Thassilo: Librosa/librosa: 0.9.1. https://doi.org/10.5281/zenodo.6097378. https://doi.org/10.5281/zenodo.6097378

[45] Li, R., Yang, S., Ross, D.A., Kanazawa, A.: Learn to Dance with AIST++: Music Conditioned 3D Dance Generation (2021)