# Automated Dance Choreography and Direction Using Music Classification and Contextual Factors

Vidisha Chandra
Computer Science and
Engineering Department
PES University
*Bengaluru, India*
vidishasateesh@gmail.com

Vibha Murthy
Computer Science and
Engineering Department
PES University
*Bengaluru, India*
vibha.harsha@gmail.com

Vishakha Hegde
Computer Science and
Engineering Department
PES University
*Bengaluru, India*
vishakhahegde23@gmail.com

Rayudu Srishti
Computer Science and
Engineering Department
PES University
*Bengaluru, India*
rayudu.srishti7@gmail.com

K S Srinivas
Computer Science and
Engineering Department
PES University
*Bengaluru, India*
srinivasks@pes.edu

***Abstract*—Dance is an artform that involves body movements, facial expressions, and gestures as a means to communicate and express the individual's thoughts and emotions while connecting with others in a non-verbal manner. As an expression of human sentiment, dance has often been used as an avenue to portray these emotions and tell a story using our bodies. Most dancers and choreographers spend lots of time, money and effort while choreographing dances. Moreover, creating dance pieces that capture the emotion of the music and match the song's beats and tempo is a challenging task. Creating unique and meaningful dance pieces that complement the music and convey the desired emotions and story is important for the best viewer experience. In this research paper, we aim to explore the interconnections between music emotion and dance motion, and simplify the choreographing dance process. For this, we use audio analysis and emotion extraction, a cross-modal transformer to generate motion embeddings, a 3D human mesh figure to display the final dance, and stable diffusion models to generate costumes and backgrounds for the dance. The goal is to create a visually pleasing, immersive experience that accurately represents the input audio's musical characteristics. Our results can be used by choreographers and dancers to design and enhance their performance routines.**

***Keywords—Transformer, Stable Diffusion, LLM, Dance generation***

## I. INTRODUCTION

In the realm of artistic expression, the convergence of music and dance has long been a captivating avenue for storytelling and emotional resonance. This paper introduces a practical and innovative pipeline that merges machine learning and generative techniques to enhance the choreographic process. The initial step employs the k-nearest neighbors (KNN) algorithm to classify a given music piece, establishing its genre as a foundational element. Moving beyond genre identification, we extract nuanced emotional elements from the music, providing a fine understanding of its affective qualities.

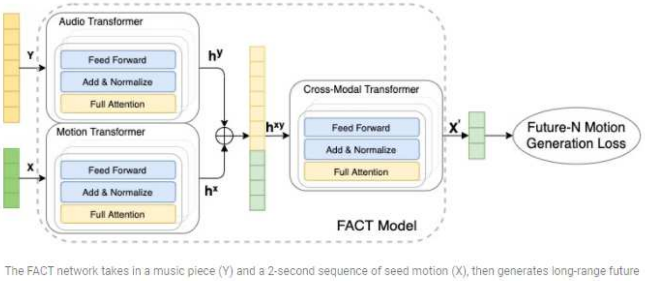
With this dual understanding of genre and emotion, we construct a prompt that guides the generation of keywords for dance costumes. These keywords serve as inputs for stable diffusion methods to create visually cohesive costume designs. This approach not only streamlines the costume design process but also ensures a harmonious connection between the visual elements and the emotional tone of the music. Expanding the scope to set design, the identified genre acts as a prompt to generate a background image that complements the dance performance. This contextual backdrop enhances the thematic unity of the performance, creating a seamless integration of visual and auditory elements.

The integration of genre and emotion, along with costume and background generation is what gives our solution an edge over existing ones. This paper looks at existing solutions in the literature review, followed by the dataset we used for training our models. Next, we describe the general workflow and implementation followed by the results that we got.

## II. LITERATURE REVIEW

### *A.* [1] FACT

*Summary:* "Dancing is a universal language found in all cultures, and generating realistic 3D dance motion from music is a challenging task". In this work, a "Full Attention Cross-modal Transformer" ("FACT") network is proposed to address these challenges. The model generates a “long sequence of realistic 3D dance motions from a short seed motion and audio features”. “FACT” involves “three key design choices”, including "full-attention mask," "N future motion prediction," and "cross-modal transformer." The “motion quality, generation diversity, and motion-music correlation” of the generated 3D dance motion are evaluated using "Frechet Inception Distance" (FID), "average Euclidean distance," and "Beat Alignment Score," respectively.



The FACT network takes in a music piece (Y) and a 2-second sequence of seed motion (X), then generates long-range future motions that correlate with the input music.

Figure 1. Full Attention Cross Modal Transformer model

*Methodology:* The proposed “FACT” model “encodes the seed motion and audio features” using a "motion transformer" and "audio transformer" into motion and audio embeddings, respectively. These embeddings are concatenated and sent to a "cross-modal transformer" that generates” N future motion sequences”. The model is trained " self-supervised” and applied in an auto-regressive framework at test time. “FACT” uses a "full-attention mask" and “predicts N future motions” beyond the current input to pay more attention to the “temporal context”.

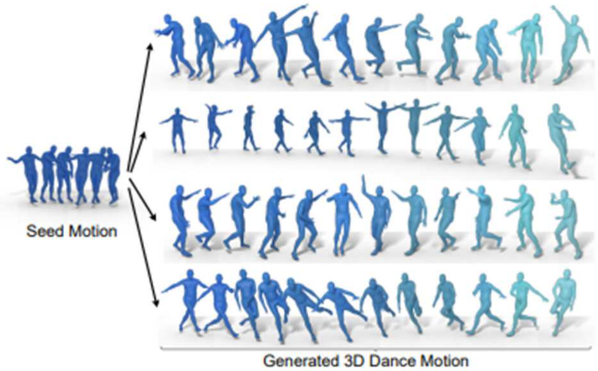


Figure 2. Generated choreography with initial seed motion

*Evaluation:* The generated 3D dance motion is evaluated using FID, “average Euclidean distance”, and “Beat Alignment Score”. The “motion quality, generation diversity, and motion-music correlation” of the generated motion are compared with three baselines, including “GPT style causal transformer, motion encoder-decoder, and PCA-based method”. The evaluation results show that the proposed “FACT” model generates 3D dance motion with “high motion quality, generation diversity, and motion-music correlation”.

$$\text{Beat Align} = \frac{1}{m}\sum_{i=1}^{m} \exp\left(-\frac{min_{\forall t_j^y \epsilon B^y}\left\| t_i^x - t_j^y \right\|^2}{2\sigma^2}\right) \quad (1)$$

*Shortcomings:* The proposed “FACT” model does not reason about "physical interactions between the dancer and the floor”, “leading to artifacts such as foot sliding and floating”. Also, the model is currently "deterministic," and generating “multiple realistic dance motions per music is an exciting direction for future research.”

## *B.* [2] A Survey of Diffusion Based Image Generation Models

*Summary:* The literature reviews diffusion-based models, focusing on stable diffusion for image generation. Stable diffusion aims to iteratively remove noise to generate high-quality images through forward and reverse processes.

*Methodology:* Forward and Reverse Process Modeling: Stable diffusion mathematically models noise addition and removal with determined diffusion rates. Components: Utilized a noise prediction module (U-net or transformer), condition encoders (T5, CLIP), super-resolution modules, and dimension reduction techniques. Subject-Driven Generation: Extended stable diffusion to allow user-defined concepts with special tokens and fine-tuning.

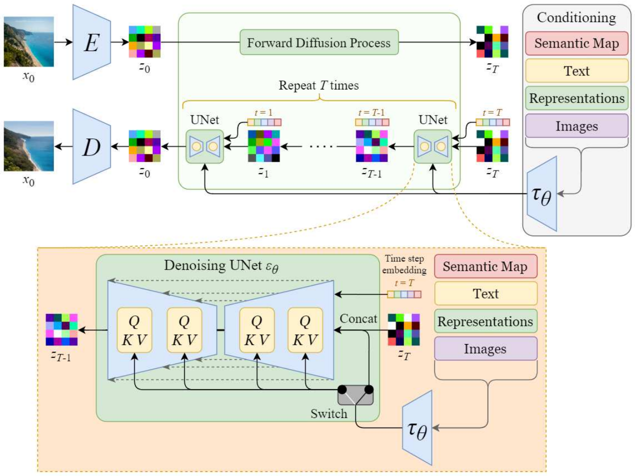


Figure 3. Diffusion process for image generation

*Evaluation:* Specific metrics not mentioned, but focus on generating photorealistic images suggested FID, Inception Score, or human preference metrics.

*Shortcomings:*
Quality Improvement: Challenged in generating high-quality images addressed through MOE, instruction tuning, sampling improvements, and self-attention guidance.

Prompt Rewrite: Acknowledged the importance of detailed prompts; Promptist suggests fine-tuning language models for prompt rewriting.

Challenges and Future Directions: Identified challenges in positional generation, concept customization quality, and content appropriateness. Future directions included addressing these challenges and refining inference time.

Aim and Considerations: Aligned with using stable diffusion for prompt-based image generation, emphasizing high-quality outputs. Implemented methodologies discussed, including noise prediction, condition encoding, and subject-driven generation. Address challenges, explore prompt rewriting, and adopt suitable evaluation metrics.

### *C.* [3] Emotion extraction and recognition from music

Researchers of [3] describe a method for detecting musical emotions by extracting the features from the audio from "two channels of music recordings and employing EEG-based characteristics". On the "APM Music dataset", the suggested technique obtains an accuracy rate of 83.29%, exceeding prior work on the same dataset by Gao et al. The researchers discovered that using feature selection and machine learning can boost the accuracy of musical emotion recognition systems. The suggested system collects audio properties from both channels of music recordings, such as "RMSenergy, MFCCs, ZCR, F0, and Voiceprob". These features were developed in response to the "INTERSPEECH 2009 Emotion Challenge" and have been shown to reflect the most emotive aspects of the audio stream.

EEG-based features are also extracted utilizing an EEG-literature-based feature extraction approach. The random forest classifier is used to group all of these features for classification. The classifier can handle high-dimensional data and estimate the missing portion of the sample, which enhances prediction accuracy. The system proposed by the researchers achieved an 83.29% accuracy which was way more than what Gao et al. got upon working on the same dataset. The researchers concluded the approach of using machine learning and feature collection could achieve excellent results in terms of accuracy for musical emotion recognition.

Though the researchers of this paper were able to achieve good accuracy, their proposed system has some limitations. With only 66.8% accuracy, EEG-based signal analysis features underperformed. One of the other reasons for this is that the model was trained only on one dataset so the scope of their research was restricted to only one collection of audio.

## III. DATASET

We are using the AIST++ dataset to train the seed motion classifier. It is the largest known dance dataset and consists of 1408 dance sequences along with music.

We have 60 music clips ready that are classified into 10 music/ dance genres. The motion clips are between 7 seconds and 48 seconds. The 60 audios are different in terms of tempo and genres, and have multiple motion sequences relevant to a particular music clip.

For our implementation we have decided to use the SMPL format of motion sequences to represent the dance seed motions. The format consists of the following parameters:

Poses - (N,24,3) are the pose parameters
Trans - (N,3) is the 3D motion trajectory.
Scaling - (N,1) is for the body scaling factor.
The dataset has been split into train, validation and test sets.

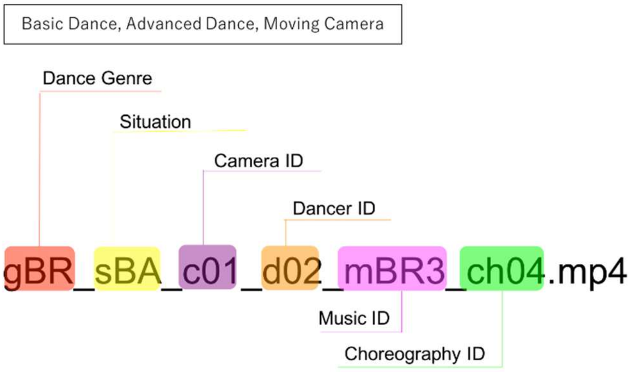

Basic Dance, Advanced Dance, Moving Camera
Dance Genre
Situation
Camera ID
Dancer ID
gBR_sBA_c01_d02_mBR3_ch04.mp4
Music ID
Choreography ID

Figure 4. Naming convention of motion sequences

## IV. WORKFLOW AND MODELS

### *4.1 Segmentation of Music Based on Emotion*

The first part of our system involves extracting music features from the input audio to break the song into segments based on emotional content. Features such as tempo, rhythm, tone and beats are extracted to analyze the song. Moreover, we have given extra importance to the emotions present in the music since we would like to take into consideration the mood and feel of the audio as well while generating dance.

We extract several music features such as length, danceability, accousticness, energy, instrumentalness, liveliness, valence, loudness, speechiness and tempo.

We train a Neural Network to classify the music into 6 emotions - Energetic, Aggressive, Sad, Happy, Calm and Dark.

Our approach involves breaking down the music into 2-second segments and classifying the segment into an emotion. Further consecutive parts of the segments which are classified into the same emotion are clubbed into segments and the indices of these segments are stored.

### *4.2 Seed Motion Selection based on Music Features*

After having analyzed the given input music and its corresponding emotions, an appropriate dance form is chosen and a dance piece is generated. We focus on mapping the audio features to human motion features that are of a certain dance style to choose the best style that will match the emotion. Seed motions play a large role in the generation of dances when using the FACT model.

TABLE I: Comparison of choreography models

| Solution | Motion Quality | Motion Quality | Motion Diversity | Motion-Music Corr |
|---|---|---|---|---|
| | FIDk | FIDg | Dist. | Beat Align |
| **Li et al. [1]** | 86.43 | 6.85 | 6.85 | .232 |
| **DanceNet [6]** | 69.18 | 2.86 | 2.86 | .232 |
| **Dance Revolution [5]** | 73.42 | 3.52 | 3.52 | .220 |
| **FACT [1]** | 35.35 | 5.94 | 5.94 | .241 |

The baseline model that is being used as a tool to generate the dance is the "Full Attention Cross Model Transformer (FACT)" [1] due to its superior performance illustrated in TABLE I. The generated dance is in the form of a 3D figure which performs the routine. The most realistic approach to show this would be in a form that replicates the movements of the human body. To provide an immersive experience, the choreographed routine that is generated must be in rhythm with the beats of the provided music.

The Seed motion selection can be further divided into the following steps:

Like every other project, the first step in our system is the creation of a data set with audio and motion features. The mentioned features are combined in the form of a tuple (audio_features, motion_features). The audio features mentioned above are concatenated to a one hot encoding vector containing the emotion classification information.

As the seed motion majorly influences the dance that would be generated for the classified emotion, a model must be trained to do so. We use a Support Vector Machine as the model for this purpose. Given the audio features as input, our model would give out motion features as output. The format of the data set is [(audio_features_1,motion_features_1), (audio_features_2, motion_features_2), ...].

A Support Vector Machine is a supervised learning algorithm that is useful for classification tasks. The goal is to find the hyperplane that divides the data set into different classes, as well as maximize the margins between them.

SVMs can also use kernel tricks to transform higher dimensional data points to a lower dimensional space where in, the data becomes separable in a linear manner.

$$Margin = (x_1 - x_2). \frac{w}{\|w\|} = \frac{2}{w}$$

$$\begin{aligned} & y_i(w.x_i + b) \geq 1, \quad for\ i = 1,2, \ldots, n \\ & \quad where\ y_i \in \{-1, +1\}, \\ & \quad x_i \in \mathbb{R}^d, \\ & \quad w \in \mathbb{R}^d, \\ & \quad b \in \mathbb{R} \end{aligned} \tag{2}$$

Finally, we compare the predicted motion features to existing motion features in the seed motion dataset and choose the one that is most appropriate.

This approach lets us choose appropriate seed motion to generate dances conditioned to emotions and allows the dance that is generated to be more diverse. The seed motion is injected in the beginning of each emotion segment and the relevant dance is generated per emotion.

Similarity of the predicted motion features to the existing motion features in the seed motion database is based on Cosine Similarity Scores between the 2 vectors.

The formula is as follows:

$$cosinesimilarity = \cos\theta = \frac{a.b}{\|a\|\|b\|} \tag{3}$$

The highest similarity score feature vector is then chosen as the seed motion sequence for the particular music segment. This allows new seed motions to be included later on without having to change the structure of the project as the output is compared to the most similar existing seed motions.

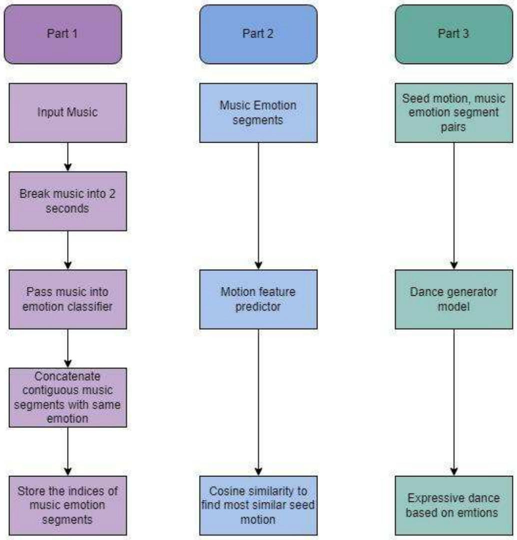

Part 1
Part 2
Part 3
Input Music
Music Emotion segments
Seed motion, music emotion segment pairs
Break music into 2 seconds
Pass music into emotion classifier
Motion feature predictor
Dance generator model
Concatenate contiguous music segments with same emotion
Store the indices of music emotion segments
Cosine similarity to find most similar seed motion
Expressive dance based on emtions

Figure 5. Workflow

### *4.3 Genre Detection*

To determine the genre of the input audio signals for this component, we used the prominent non-parametric classification technique, the K-Nearest Neighbours (KNN) algorithm. KNN was chosen for its simplicity, effectiveness, and suitability for applications where decision boundaries are complex and nonlinear.

Each element in the dataset is represented by its mean matrix, covariance matrix, and its respective class label. The class labels correspond to different music genres, providing a labeled training set for the KNN algorithm.

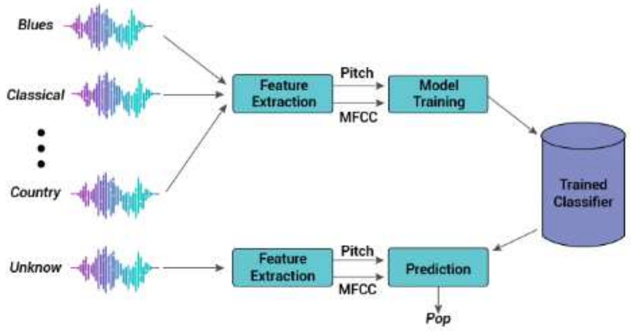


Figure 6. Genre classification using MFCCs and KNN model

For the input audio, we extracted Mel-Frequency Cepstral Coefficients (MFCCs) using the librosa library. These coefficients capture the spectral characteristics of the audio signal and serve as fundamental features for genre classification. The mean and covariance matrices of the MFCCs were computed and used as input features for the KNN algorithm.

The Mahalanobis distance metric is employed to calculate the dissimilarity between instances in the feature space. The distance is calculated as follows:

$$D_M(X,Y)=(X-Y)T\cdot S-1\cdot(X-Y) \qquad (4)$$

Where, X and Y are column vectors representing the two points. S is the covariance matrix of the distribution. This distance metric considers both the mean and covariance information, providing a robust measure of similarity between audio feature representations.

Then the k-nearest neighbors in the training set are recognised. The class labels of these neighbors undergo a voting process for the nearest class, determining the predicted genre for the input audio.

This detected genre is then used further in the costume and background pipelines.

While KNN provides a direct and interpretable approach to music genre detection, there are certain considerations. The choice of the parameter k, representing the number of neighbors, can impact classification accuracy. Further research could explore optimization techniques for determining an optimal k value. Additionally, normalizing features and investigating other distance metrics may enhance the model's performance in diverse audio environments.

### *4.4 Costume Generation*

In the context of costume generation, our methodology revolves around the nuanced interpretation of two fundamental components inherent in music—namely, emotion and genre. These are obtained after a thorough analysis of music features such as beat and tempo. The generation of costumes involves two distinct phases. Firstly, we use a large language model (LLM) to create a prompt that encapsulates keywords of the outfits and accessories required for the dance, taking into consideration the emotion and genre of the song. Subsequently, this curated prompt is seamlessly integrated into a stable diffusion model specifically engineered for text-to-image generation, resulting in an appropriate costume. This approach ensures that the costume matches the music characteristics accurately and highlights the effectiveness of our method.

### *4.5 Background Generation*

Background generation is done by taking into account the genre of the song alone. The Cohere LLM is used to create the prompt describing an appropriate dance setting fit for the mood and tonality of the song. The prompt is then passed into the Stability stable diffusion model which generates an image of a possible background the user could incorporate into their dance routine.

Cohere is a large language model that specializes in generating coherent and contextually relevant text based on given prompts. It excels in natural language understanding, making it valuable for tasks such as creative writing. The Stability model focuses on image stability and generation. It likely leverages advanced algorithms to generate stable and visually coherent images based on textual prompts.

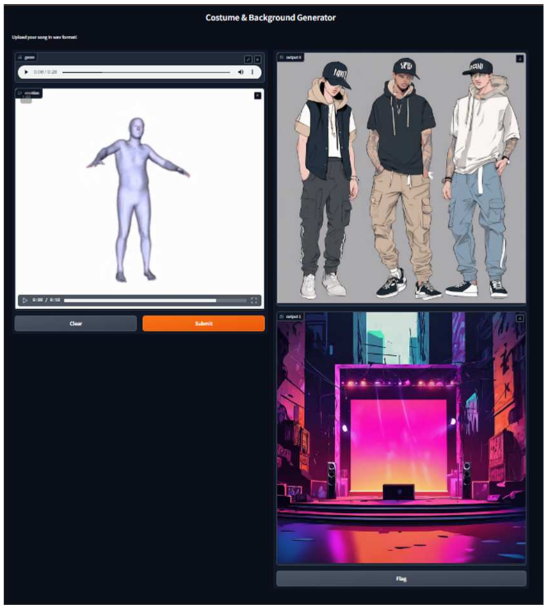

Costume & Background Generator
Upload your song in wav format:
genre
0:06 / 0:28
output 0
output 1
Clear
Submit
Flag

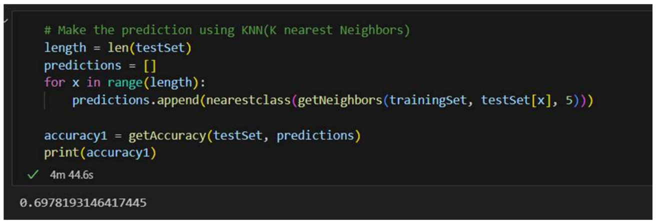
Figure 7. Generated costume and background

## V. RESULTS

*Genre Detection KNN accuracy:*

The accuracy is calculated by dividing the count of correctly predicted instances by the total number of instances in the test set. The counter is incremented for each instance where the predicted genre label matches the actual label. Multiplying by 1.0 ensures the result is a floating-point number. The formula for the same is:

Accuracy = Total Number of Instances in the Test Set / Number of Correct Predictions (5)



```
# Make the prediction using KNN(K nearest Neighbors)
length = len(testSet)
predictions = []
for x in range(length):
    predictions.append(nearestclass(getNeighbors(trainingSet, testSet[x], 5)))

accuracy1 = getAccuracy(testSet, predictions)
print(accuracy1)
```

✓ 4m 44.6s

```
0.6978193146417445
```

Figure 8. Accuracy of the KNN model

Audio features have been extracted from various sources to see how the model performs on all types of songs. An emotion classifier has been implemented using a neural network that can accurately classify audio data into six distinct emotion classes (Energetic, Relaxing, Dark, Aggressive, Sad, Happy). We obtained an accuracy of 0.69

*Emotion Detection Neural Network accuracy:*

We used the accuracy measure to evaluate the working of the neural network model in the classification of music to emotion and obtained an accuracy of 0.72.

*Seed motion selection Hinge loss:*

Furthermore, SVM is used to choose seed motion for the emotion appropriately. Using the baseline FACT model, we have generated multiple dances for each song to see the uniqueness of each dance generated through a different seed motion each time. Below is an example of how a particular audio is split into subsongs based on emotion and the corresponding human 3D mesh animation for this:

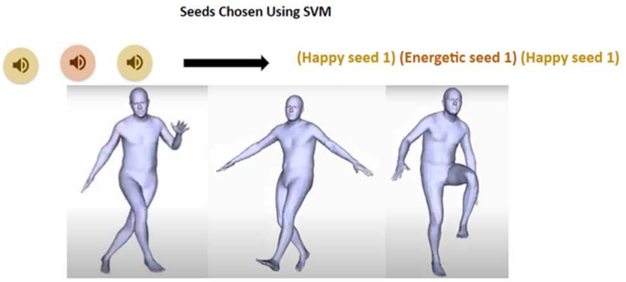


Figure 9. Seed motion selection based on emotion

To measure the accuracy for mapping the seed motions, we used the Hinge Loss function is defined as:

$$L(y, f(x)) = max(0, 1 - y * f(x)) \quad (6)$$

The Hinge Loss function uses the proportion of the distance from the SVM margin as a measure of accuracy of classification. The objective is to minimize Hinge Loss. A well-classified SVM would usually have hinge loss values close to or equal to 0 for the correct class and positive hinge loss values for the incorrect classes.

$$Objective = \Sigma\, L(yi, f(xi)) + \lambda * ||w||^2 \quad (7)$$

We got an average Hinge Loss of 0.57 after testing on different song genres and emotions.

FID = 〚||μ_r - μ_g||〛^2 + Tr(Σ_r + Σ_g - 2 〚(Σ_rΣ_g)〛^(1/2)) (8)

## VI. USER STUDY

We conducted a user study to assess the fluidity and non-repetition of the dance sequences generated by our model by comparing it with other existing models. Over 35 participants were asked to rate each dance sequence on a scale of 1 to 5. Table II depicts the results of the study. Our dance sequences were commended for their avoidance of repeated steps, indicating the diversity and richness of the choreography produced by our model.

200 image–music pairs were chosen for the user study and evaluation. The users were given the task of determining whether they felt the background and costume image generated matched the given music or not. The ratings were taken as 2 for 'Yes', 1 for 'Somewhat', and 0 for 'No. This gave us an insight into how accurately the emotion and genre detection models performed and how appropriately the prompts were constructed. To evaluate the image quality, we provided the users with pairs of model-generated costumes and backgrounds versus ground truth images that matched the music. The percentage of times that the user chose the model-generated images over the ground truth images was calculated.Average ratings for image-music alignment and percentage preference of generated images are reported.

TABLE II: Result of user studies for FACT with seed motion introduction vs original FACT

| Solution | Fluidity | Non - Repetition |
|---|---|---|
| **FACT [1]** | 3.8 | 2.5 |
| **Our Model** | 3.9 | 3.2 |

Table III: Result of user studies for background and costume generation

| **Generated image** | **Image-music alignment (Average rating)** | **Image quality (Percentage %)** |
|---|---|---|
| **Background** | 1.82 | 68 |
| **Costume** | 1.69 | 57 |

## VII. CONCLUSION

Dance relies heavily on the genre and emotions of the music, helping viewers connect with the performance. By focusing on both the genre and emotion of the music, not just the beats, our study achieves better results compared to existing solutions. Selecting appropriate seed motions based on emotion further refines the choreography. Additionally, the setting and costumes play a crucial role in full-scale choreography. By incorporating elements such as music genre, emotion, seed motion, costume, and background, we provide a comprehensive toolkit for both professional and beginner dancers.

Currently, our limitations include the inability to visualize costumes and backgrounds on the 3D mesh figure and the need for a wider pool of seed motions to improve the steps. At present, our steps are generated for a single dancer. In the future, we aim to expand our scope to include formations and steps for group performances, enhancing the choreography for multiple dancers. Additionally, we aim to expand the dance styles to include not only Western forms but also Indian classical dance forms.

## VIII. ACKNOWLEDGEMENT

We would like to express our gratitude to the Department of Computer Science and Engineering, PES University, for giving us this opportunity. The guidance, assistance, and encouragement we received throughout the development of this project was invaluable. Finally, this project could not have been completed without the continual support and encouragement we have received from our family and friends.

## IX. REFERENCES

[1] R. Li, S. Yang, D. A. Ross, A. Kanazawa, AI Choreographer: Music Conditioned 3D Dance Generation with AIST++ vis ICCV 2021

[2] Zhang, Tianyi, Zheng Wang, Jing Huang, Mohiuddin Muhammad Tasnim, and Wei Shi. "A Survey of Diffusion Based Image Generation Models: Issues and Their Solutions." arXiv preprint arXiv:2308.13142 (2023).

[3] Zhang F, Meng H, Li M. Emotion extraction and recognition from music. 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)

[4] Li Siyao, Weijiang Yu, Tianpei Gu, Chunze Lin, Quan Wang, Chen Qian, Chen Change Loy, Ziwei Liu, Bailando: 3D Dance Generation by Actor-Critic GPT with Choreographic Memory Mar 2022

[5] Jonathan Tseng, Rodrigo Castellon, C. Karen Liu Stanford University, EDGE: Editable Dance Generation From Music, Nov 2022

[6] Nawaz R, Nisar H, Voon YV, Yee TP. Acoustic Feature Extraction from Music Songs to Predict Emotions Using Neural Networks. 2018 2nd International Conference on BioSignal Analysis, Processing and Systems (ICBAPS)

[7] Zhang F, Meng H, Li M. Emotion extraction and recognition from music. 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)

[8] Au, Ho Yin, Jie Chen, Junkun Jiang, and Yike Guo. ChoreoGraph: Music-Conditioned Automatic Dance Choreography over a Style and Tempo Consistent Dynamic Graph, Jul 2022

[9] Kim, Jinwoo, Heeseok Oh, Seongjean Kim, Hoseok Tong, and Sanghoon Lee. A Brand New Dance Partner: Music-Conditioned Pluralistic Dancing Controlled by Multiple Dance Genres, 2022.