Dance Step Recommender

Chinmay Kulkarni
Department of Computer Science and Engineering
Government College of Engineering, Aurangabad

Ritesh Nemade
Department of Computer Science and Engineering
Government College of Engineering, Aurangabad

Sanket Sutar
Department of Computer Science and Engineering
Government College of Engineering, Aurangabad

Akshay Tarde
Department of Computer Science and Engineering
Government College of Engineering, Aurangabad

Mandar Deshpande
Department of Computer Science and Engineering
Government College of Engineering, Aurangabad

Madhuri Joshi, PhD
Department of Computer Science and Engineering
Government College of Engineering, Aurangabad

ABSTRACT

The huge measure of information accessible on the Internet has prompted the improvement of clever frameworks. Lately, there has been a ton of consideration in frameworks like music recommender, book recommender, film recommender, and so on. Likewise, the motivation behind this project is to use soft computing techniques to develop a choreography system. This article outlines a detailed summary of the "Dance Choreography System". The system choreographs dance by recommending dance steps, considering the time signature and tempo of the given soundtrack as its core components. The paper incorporates a portrayal of the subject, framework engineering, and gives an itemized depiction of the work done to point. The project outputs the dance steps, applied to an animated character that is displayed on the browser.

Keywords

Time Signature, Tempo, Dance Choreography, Animation.

1. INTRODUCTION

Dance Choreography: Most of the features of a soundtrack that are required for dance choreography are tempo and time signature [1]. The time signature of a song is the number of beats in a bar. For designing the dance choreography system, the Spotify database was used to get the required details of the song. Some additional features like danceability, valence, and energy are also used for better outcomes [2]. The features are defined as follows:

Time Signature: The time signature (meter) is a term that defines the number of beats in each bar (or measure). Tempo: The overall estimated tempo of a track is in beats per minute (BPM). In musical terminology, the tempo is the speed or pace of a given piece and derives directly from the average beat duration.

Danceability: Danceability determines how good a dance track is based on a variety of musical elements like tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.

Valence: A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry)

Energy: Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale. Perceptual features contributing to this attribute include dynamic range, perceived loudness, timbre, onset rate, and general entropy.

2. PROPOSED MODEL

2.1 Algorithm

- Input song unique ID
- Obtaining the required features of a song using Spotify API (Song unique ID)
- Selecting the required number of dance steps that correspond with the time signature of the song
- Applying animations to the character
- Combining the dance steps. This can be well understood in Fig 1.

2.2 Database

The following databases are being used for the system:

2.2.1 Spotify

It is a digital music service that gives access to millions of songs. It can be used to access the database of the songs. It provides its own Web API. Developers can utilize this Spotify API to retrieve Spotify content such as album data and playlist. Also one can get information about songs like Tempo, Time Signature, Energy, Acousticness, Liveliness, and many more [3].

2.2.2 Mixamo

It is a 3D graphics technology company. It is owned by Adobe. It provides a database of hundreds of animations available in different formats like fbx, glb, bvh, etc. The dance steps that the system is using are obtained from Mixamo [4].

2.2.3 Animation Metadata

The metadata of the animation database has the following attributes: Time Signature: The time signature for which the dance steps are choreographed.

Natural Tempo: The natural tempo of a step is the tempo for which the step best suited.

Dance form: This attribute corresponds with the dance form of the step.

Flexibility: The flexibility is the measure of the stretchability of the step. It is introduced to preserve the elegance of a dance step without compromising the accuracy. The flexibility depends on the defined flexibility factor. The flexibility factor is the measure of how stretchable the dance steps are i.e. if a dance step is choreographed for a tempo than what is the range of tempos in which the step will neither be overstretched nor over shrunk. The flexibility factor lies between 0 and 1, both inclusive.

Let,

The flexibility factor is δ

The natural tempo is t

The lower limit of the flexibility of the tempo is L.

The upper limit of the flexibility of the tempo is U.

Then the L is given by:

$$L = (1 - \delta) \times t$$

And U is given by:

$$U = \frac{1}{(1 - \delta)} \times t$$

2.3 Tools

The following softwares have been used for the development of the system:

2.3.1 Python

Python being an easy to write language finds its roots everywhere, even in Spotify. The support that Python provides in various mediums makes it comfortable to work with and also helps in integrating the system effectively.

2.3.2 Blender

Blender is the 3D creation suite that is free and open source. It supports the entire modeling, rigging, animation, simulation, rendering, compositing and motion tracking of the 3D pipeline, including video editing and development of games.

2.3.3 THREE.js

Three.js is a first mainstream JavaScript structure for showing 3D content on the web, giving the ability to show extraordinary models, games, music recordings, logical and information perceptions, or basically anything one can envision, directly in their browser and on the Smartphone [5][6].

2.3.4 Flask

Flask is a lightweight WSGI web application structure. It is intended to make beginning brisk and simple, with the capacity to scale up to complex applications. It started as a basic covering around Werkzeug and Jinja and has gotten one of the most famous Python web application structures. Flask offers proposals however doesn't implement any conditions or venture design. It is dependent upon the designer to pick the devices and libraries they need to utilize. There are many extensions provided by the community that makes adding new functionality easy.

2.3.5 JavaScript (React)

React is a User Interface JavaScript library created and developed by Facebook. React is an amazing asset for building UI components.

2.4 Feedback Mechanism

This recommender system is brilliant enough as it continually develops itself. The Dance Recommender system improves itself based on two simultaneous feedback mechanisms [7][8]. The system takes feedback from the user on the choreographed dance video and also indirectly monitors user's preferences thus suggesting more suitable dance steps.

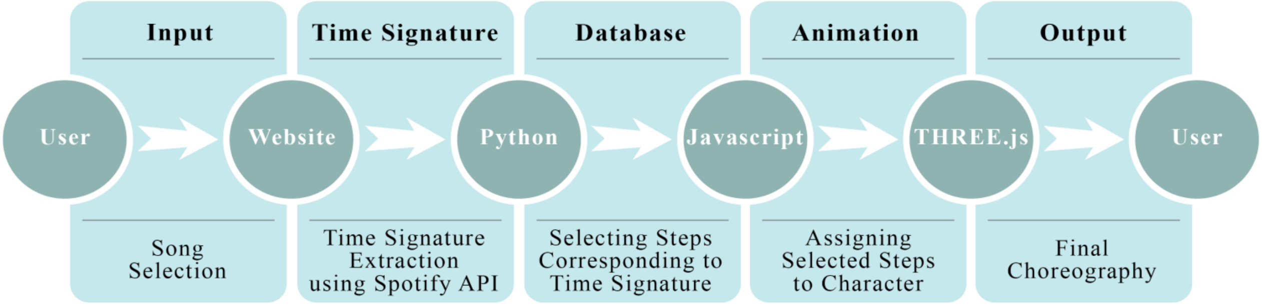
Input
Time Signature
Database
Animation
Output
User
Website
Python
Javascript
THREE.js
User
Song Selection
Time Signature Extraction using Spotify API
Selecting Steps Corresponding to Time Signature
Assigning Selected Steps to Character
Final Choreography

Fig 1: Algorithm Flowchart

3. EXPERIMENTAL ANALYSIS

To find the appropriate value of the flexibility factor for the system, an experiment was conducted in which 25 participants were involved. Five musical tracks having different tempos labeled Track01, Track02, Track03, Track04, Track05 with natural tempos 80, 100, and 120 respectively, were selected individually and discrete values of flexibility factor were applied to the natural tempo. The participants then assessed the suitability and the number of choices they had. The number of choices had to be maximized keeping into account the user experience.

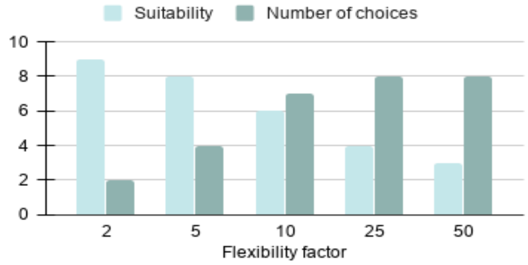

Fig 2(a): Tempo 80 bpm

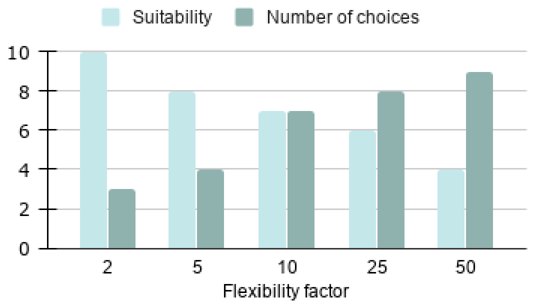

Fig 2(b): Tempo 100 bpm

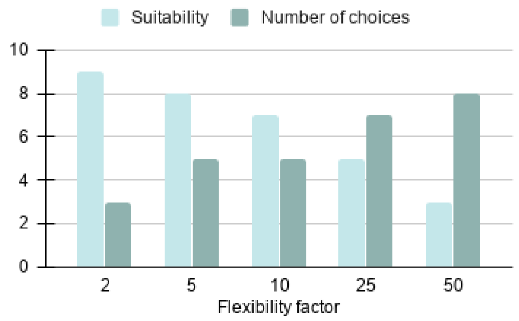

Fig 2(c): Tempo 120 bpm

Fig 2: User response on a scale of 0-10 with respect to the flexibility factor

4. RELATED WORK

Ubisoft Just Dance: Just Dance is a rhythm game series developed and published by Ubisoft [9]. *Just Dance* is a motion-based dancing game for multiple players, with each game including a collection of songs each with their own dance choreographies. For each track, choreographies are pre-decided by Ubisoft and the players cannot change it according to their will. Thus the player can only perform on given steps. During each song, players mirror a dance performed by actors on the screen, following commands that appear on the screen, and are awarded for their accuracy. The actor on the screen is nothing but a character developed by Ubisoft. Depending on the game and system it is played on, the game can be played with either motion controllers or cameras.

There are some key differences between Just Dance and this Dance Step Recommender system. First of all, Just Dance requires a motion controller while the recommender system doesn't require any. Just Dance provides only one choreographed dance to the user for the chosen track. But in this Dance Recommender system, the user is provided with a large number of dance steps and can choose according to their requirement and interest. The use of these two systems also varies. As Just Dance is a game and the user tries to score the maximum number of points the dance video has to be played till the end once it begins. On the other hand, this system is used to learn the dance on a music track, and hence the user will rewind the video on numerous occasions till achieving flawlessness.

5. RESULTS

Various dance steps are being concatenated to form animation sequence [10] and this is shown in the form of 3D character animation. The platform which is being designed is highly efficient than the existing frameworks.

The mathematics of the algorithm is as follows:

Let,

The number of suitable steps for the song in the database is **n**.

The number of required steps for the input song is **r**.

Therefore, the number of ways in which r steps can be selected from n = $^{\mathbf{n}}\mathbf{C}_{\mathbf{r}}$

The number of ways r steps can be arranged = **r!**

Therefore, the number of ways a choreography can be done = $^{\mathbf{n}}\mathbf{C}_{\mathbf{r}}\mathbf{r!} = {}^{\mathbf{n}}\mathbf{P}_{\mathbf{r}}$

On putting the values r = 25 (average steps required for a song) and n = 50 (total number of suitable steps for the song in the system) the number of ways choreography is done is approximately $^{\mathbf{50}}\mathbf{P}_{\mathbf{25}}$.

The time difference between the expected time of the end of the last dance step and its actual position is the latency. The accuracy then is calculated as follows:

Let,

The duration of the song is *t*.

The latency is θ

Then the accuracy is given by

$$\text{Accuracy} = \frac{t-\theta}{t}$$

Along with accuracy some of the evaluation metrics that are used for the performance analysis of the recommender system are:

User preference: Of the total recommendations the system provides to users they are asked to choose how many recommendations they felt were suitable for them. Averaging this number of suitable recommendations of all those users we get the metric User preference. This experiment was carried out on a group of 25 participants and the user preference was found to be 71.24%.

Coverage: Coverage is a metric that determines how many things (items, users, or ratings) the recommender system is able to recommend out of the total recommendable item set. Shani et al.[11] describes catalog coverage as the proportion of items that the recommendation system can recommend. Item coverage is the margin of items that get included in the recommendation list over the number of potential items. In this system, Item coverage can be used to evaluate the number of dance steps that are being recommended out of the available set.

Novelty: Novelty is an evaluation metric that looks for novel items in the recommendations made to the user. A novel item is the one that the user hasn't yet interacted with. Zhou et al.[12] explains novelty as a generation of unexpected results by the recommender system. This turns out to be useful for the evaluation of the recommender system as more the number of novel items it will recommend, more the user will be exposed to new content. In regards to the Dance Step Recommender system, the novelty will be eyeing on how many different and unknown steps have been recommended to the users and ultimately enriching their experience of the system.

Serendipity: Serendipity is a metric which is a measure of how surprising and pleasant a recommended item is to the user. Zhang et al.[13] says that serendipity is "unusualness" or "surprise" in recommendations. There are several definitions of serendipity as an evaluation metric but almost all of them agree that the surprising element is useful to the user. After looking into novelty and serendipity one may find them almost the same. But there's a fine line between both of them. Novelty only determines how unknown the recommended item was but serendipity also takes into consideration its relevance for the user.

The animated character created in Mixamo performing the dance moves is shown in Fig 3.

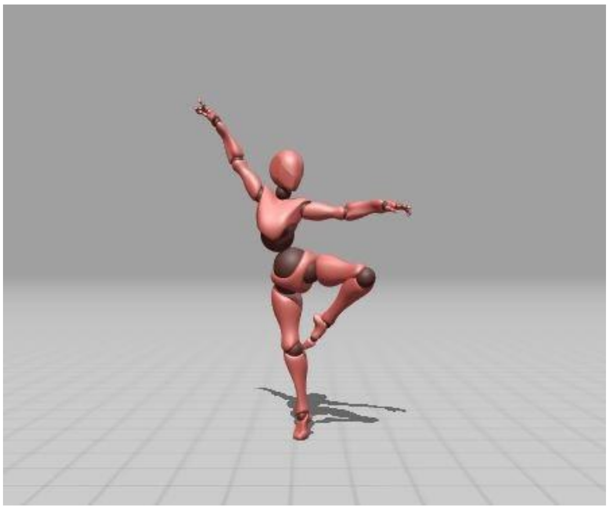

Fig 3: Character Performing Dance Step

6. CONCLUSION

Today the world is witnessing unavailability of a smart choreography system that can help the amateurs learn to dance. To overcome this hurdle we designed this Dance choreography system. The system mainly performs the task of choreographing a dance on an input soundtrack with the help of various music attributes. A choreographed dance is made by integrating various dance steps that are suitable for the input track. This dance is performed by an animated character, making it easy for the user to learn. It also provides users the freedom to choose steps from the available set and to arrange them in their own way, making it a unique output.

In the future, the consolidation of Motion Capture and Natural Language Processing will make it more efficient. The Natural Language Processing will incorporate lyrics of the input track for choreographing the dance and thus improving the accuracy of the system.

7. REFERENCES

[1] Mikel Gainza, Eugene Coyle, Vienna, Austria, 2007. Time Signature Detection by Using a Multi-Resolution Audio Similarity Matrix

[2] Spotify Audio Features: https://developer.spotify.com/documentation/web-api/reference/tracks/get-audio-features/

[3] Spotify Audio Analysis: https://developer.spotify.com/documentation/web-api/reference/tracks/get-audio-analysis/

[4] Mixamo 3D Animations: https://www.mixamo.com/#/?genres=Dance&page=1&type=Motion%2CMotionPack

[5] Jos Dirksen, Learn Three.js - Third Edition, 2018, Packt Publishing Ltd

[6] Jos Dirksen, Three.js Essentials, 2014, Packt Publishing Ltd

[7] Douglas W. Oard and Jinmook Kim, AAAI 1998, Implicit Feedback for Recommender Systems

[8] Ladislav Peska, 2016, Using the Context of User Feedback in Recommender Systems

[9] Ubisoft Just Dance: https://www.ubisoft.com/en-us/game/just-dance-2020

[10] Stefano Corazza, 2011, Real-Time Automatic Concatenation Of 3D Animation Sequences

[11] Shani G, Gunawardana A (2009) Evaluating recommender systems. Microsoft Research.

[12] Zhou T, Kuscsik Z, Liu J et al (2010) Solving the apparent diversity-accuracy dilemma of recommender systems. Proc Natl Acad Sci 107(10):4511–4515

[13] Zhang Y, Séaghdha D, Quercia D, Jambor T (2012) Auralist: introducing serendipity into music recommendation. In: WSDM’12. ACM, New York, NY, USA, pp 13–22