DanceDJ: A 3D Dance Animation Authoring System for Live Performance

Naoya Iwamoto[1⋆], Takuya Kato[1⋆], Hubert P. H. Shum[2],
Ryo Kakitsuka[1], Kenta Hara[3], Shigeo Morishima[4]

[1] Waseda University
[2] Northumbria University
[3] Meiji University
[4] Waseda Research Institute for Science and Engineering
iwamoto@toki.waseda.jp, takuya_lbj@ruri.waseda.jp,
hubert.shum@northumbria.ac.uk, kakitsuka.99821@ruri.waseda.jp,
mactkg@gmail.com, shigeo@waseda.jp

Abstract. Dance is an important component of live performance for expressing emotion and presenting visual context. Human dance performances typically require expert knowledge of dance choreography and professional rehearsal, which are too costly for casual entertainment venues and clubs. Recent advancements in character animation and motion synthesis have made it possible to synthesize virtual 3D dance characters in real-time. The major problem in existing systems is a lack of an intuitive interfaces to control the animation for real-time dance controls. We propose a new system called the *DanceDJ* to solve this problem. Our system consists of two parts. The first part is an underlying motion analysis system that evaluates motion features including dance features such as the postures and movement tempo, as well as audio features such as the music tempo and structure. As a pre-process, given a dancing motion database, our system evaluates the quality of possible timings to connect and switch different dancing motions. During run-time, we propose a control interface that provides visual guidance. We observe that disk jockeys (DJs) effectively control the mixing of music using the DJ controller, and therefore propose a DJ controller for controlling dancing characters. This allows DJs to transfer their skills from music control to dance control using a similar hardware setup. We map different motion control functions onto the DJ controller, and visualize the timing of natural connection points, such that the DJ can effectively govern the synthesized dance motion. We conducted two user experiments to evaluate the user experience and the quality of the dance character. Quantitative analysis shows that our system performs well in both motion control and simulation quality.

Keywords: Human-computer Interaction, Character Animation, DJ controllers, Dance

[⋆] The first two authors contributed equally to this work.

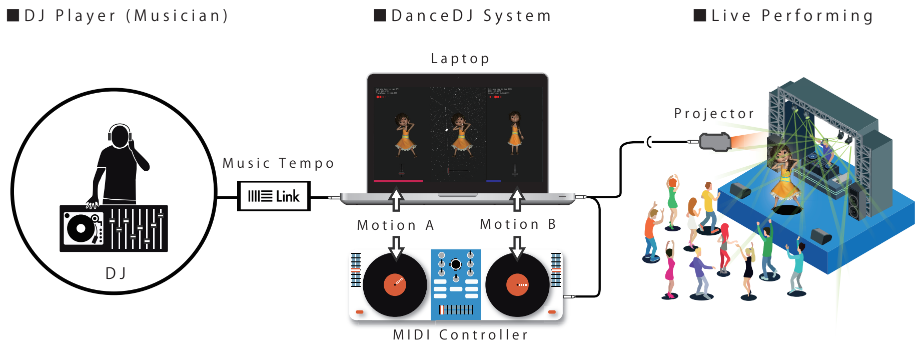


Fig. 1. System overview of DanceDJ with live performance setup. (a) DJ plays music sending the tempo into DanceDJ using local network. (b) Dance DJ receives the tempo and allows the user to synchronize. (c) Audiences can enjoy watching the virtual avatar's dance synchronized with the music on the stage/screen as a visual expression of the music.

1 Introduction

Dancing is one of the most popular physical activities across every age group, race and region around the world. It can be regarded as a form of self-expression and a means of communication. The use of dance to enhance musical expression has been popular for centuries in musicals, opera and ballet. Video of well designed dance motion can be played along a musical score to achieve a similar effects.

In recent years, dance animation using digitally generated characters has become more and more popular in musical animation mega hits such as the animation "Frozen". Apart from such a kind of *pre-rendered* dance animation, real-time dance authoring has also been introduced. Using novel sensors such as the Microsoft Kinect and motion capture systems, the real-time control of characters had become more accessible. These interfaces allow the user to create computer generated dance motions with intuitive actions. While we can enjoy various live performances acted by computer generated character in the world, most of these live performances only show the 2D video created in advance as an offline process. The difficulty of live scene authoring lies not only on the skills of the performers but also the technical complexity to achieve a high quality performance. Therefore, we propose a character control system that requires a relatively low skill level to control the character, as well as adapts well in existing live stage performance frameworks.

A challenge for real-time dance authoring is that while many dance movements are based on normal motion, dance differs in that its motion normally requires a beat. In music and dance, the beat is an audible or visual cue demarcating the division of a certain sequence. Dancers continually change their poses while maintaining the beat. The rhythm of musical content is normally characterized by a repeating sequence of beats. Dance and music have evolved together over centuries to match one another by sharing the beat. Many existing interfaces have not been able to achieve high quality dance as these principles are not taken into account.

To address this problem, we came up with a hypothesis that the creation of real-time dance can be improved by allowing the users to understand the beat of both motion and music. This hypothesis came up when observing the interaction of the disk jockey in music. A disk jockey, known as a DJ, is a person who introduces and plays music especially on the radio or at a club or live performance. Their performance often involves intricate and seamless mixing to connect one piece of music with another, which is recognized as a way of performing music without playing musical instruments. Their interaction with the music normally takes the beat of the music into account. Since both dance and music share the same principles, we anticipate that by providing a means for the user to maintain the beat when creating dance motion would result in a highly usable system.

We present a novel dance authoring interface called *DanceDJ* based on such a hypothesis. By mimicking the interaction of the musical DJ, the system allows users to control dance motion more intuitively and create high-quality dance motion in real-time. By implementing the synchronize button that synchronizes the dance motion with the music played by the other electronic instruments, the user can match the beat of the dance motion to that of the music. In contrast to the music DJ, the connectivity and beat of the dance motion tend to be more abstract than that of the music, which affects the usability when connecting the dance motion. To support the users, we have implemented a novel feature to estimate the connectivity of the dance motions. The system automatically calculates the beat of the dance by using the motion intensity to estimate the probability of the frame wise dance connectivity, which represents how well the beat of dance and music matches together. This feature helps the users to achieve a DJ-like experience when interactively creating dance motion and allows users to create, as well as realizing our hypothesis of the beat and correlation with music.

We have produced a fully usable system for real-time dance authoring. Experimental results show that high-quality dance motion can be synthesized in real-time using the proposed DJ interface. We have preformed quantitative user studies to support the usability of the system. For the user, we have evaluated how well users have been able to create high-quality dance motion intuitively. For the audience, we have evaluated how plausible the created dance motions are by looking at the dance motion results our system has created in real-time. We have also conducted tests of using our system in real-world live performances.

There are two major contributions in this paper:

- To design a novel DJ interface for real-time dance authoring based on dance and music beat correlations.
- To introduce a novel dance and music beat evaluation function that evaluates how well a pose in a dance synchronizes to that of another dance using the estimated beat information of motion and music

The rest of the paper is organized as follow. Section 2 describes the related research of this project. Section 3 describes the DJ interface we have implemented for real-time dance authoring. Section 4 explains the design of the transition function that estimate the quality of switching from one motion to another based on the beat pattern. Section 5 explains how we visualize the results of the transition function for effective dance controls. Section 6 details the user study we have preformed to support the usability of the system. Finally, Section 8 concludes and discusses the system.

2 Related Work

In this section, we introduce works related to our system. There are numerous related works for creating character motion interactively. Mainly, this work can be divided into user interaction when creating character motion and DJ-like interactions.

2.1 User Interfaces for Character Control

Numerous interfaces have been introduced to control character motion. One of the favorite ways is the performance-based approach. The interface, which uses motion capture and video of human motion has been popular in creating 3DCG animation creations [11, 15, 20, 4, 5]. Depth camera based motion sensing opens another pathway for real-time character controls [23, 16, 28, 17]. While they fulfill the demands of recreating accurate human motion, this interface relies upon the user's ability to perform the demanded motion. Additionally, editing the character motion requires other interactions, such as sketch-based interaction [2, 8, 13, 9, 3]. While these interfaces are easy to use, they do not match our demand for real-time motion creation. Interfaces that control the bones of the characters [26, 10, 27, 12], such as ones introduced by [6], allows users to create dance motion intuitively. These character-shaped interfaces that interactively deform the characters' bones allow users to create arbitrary poses. Despite its intuitiveness in creating bone shapes for a specific frame, these interfaces are not suitable for creating character motion in real-time using only two hands to control the whole body. One of the ways to create character motions is to set motion coordination keys to individual buttons like the character controls in video games [25]. This way, the user can press various buttons to control the character. While the usability of such character control would be familiar to those who play video games, the user is required to memorize the dance motions allocated to each particular button. This forces the users to either memorize a large number of button allocated motions or the designer to limit variety of the motions allowed to the user.

2.2 DJ-like Interaction in Research

Due to the creative nature of mixing, many types of research have focused upon the interaction of DJ's. The interaction of DJ's has been analyzed in various ways to realize similar applications or improve their actions. While many of these applications introduce a novel interface alternative to the traditional DJ interface, the form of DJ interfaces has not changed drastically. Accordingly, recent researchers have applied the DJ interaction to other applications to take advantage of its high usability [19, 18, 7]. Target applications which apply a DJ-like interface vary from data visualization, 3DCG visualization to Robot motion control. In these researches, sliders and turntables have been installed in various applications. They have proved that DJ-like interactions can increase usability and provide the user with a novel experience of interaction with the target application, especially when this interaction is in real-time.

Our proposed user interface is inspired by the interface called "Robot-Jockey", which was introduced by [22]. This interface introduced a novel interaction for creating dance motion of a robot using the DJ-like interaction of controlling tempo and motion in real-time. Since the primary objective of this interface is to provide the means of creating the robot's motions in real-time with intuitive interaction, their interface only needs to select very limited motions such as kicking or punching from the database. The limited selection of motions can be sufficient for a robot which only has a few actuators to control the body, while 3DCG characters have several more joints, bones and skin to consider. Accordingly, we follow the idea of controlling dance motions in real-time using a DJ-like interface while we improve the system to be sufficient to control 3DCG characters.

2.3 Implementation Requirements

Taking the shortcomings of related applications towards our potential users into account, we set our implementation requirements as follows:

Intuitive motion control:
We select and control the parameters of the character motion that are necessary for the live-action remixing of dance motions.

Intuitive interface:
Our interface is based on the MIDI interface. The MIDI interface is a well-designed, intuitive, and very popular interface for many fields and requires real-time interactions.

Works in real-time:
Our interface can create dance motions in real-time, which allows usage in live performances such as concerts.

In developing our interface, we have tried to achieve all of the goals which the works cited above have not yet completely satisfied. As related works and products have not been able to fulfill the requirements set out above, we believe that this makes our interface advantageous for application in numerous settings.

3 DanceDJ Interface

To verify our hypothesis described in Section 1, we developed the prototype of DanceDJ to author character's dance motion with a DJ interface. A simple live

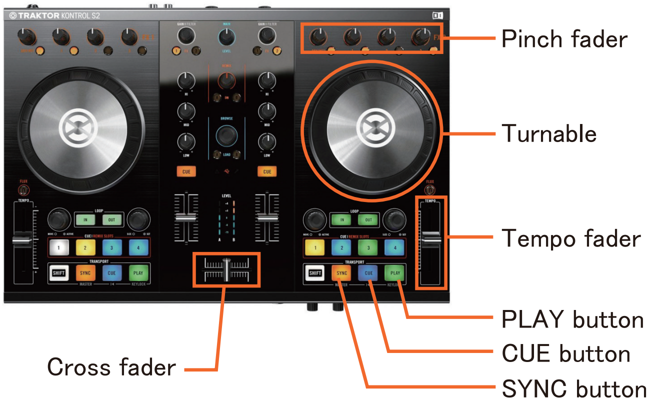


Fig. 2. Common DJ Interface; buttons, fader and turntables are equipped for mixing two music.

performance work flow is shown in Figure 1. Firstly, the DJ plays music and shares the beat with the DanceDJ system using a local network in real-time. While we evaluated our system playing only along DJ, the use of electronic instruments and existing online beat estimation systems would allow the proposed system to be applicable for wide variety of music genre. Next, the DanceDJ operator controls an input dance motion to synchronize with the received beat. Finally the character which is assigned the dance motion is projected on a screen or stage in front of audiences. From the audience's perspective, it looks like the character is dancing while synchronized with the music played by the DJ. In our experiment, the DJ used a DJ software package called *Traktor*, and sent music information to other device with a software called *Ableton Link*. The DJ interface sends a MIDI signal distributed at each button or slider into the DanceDJ system.

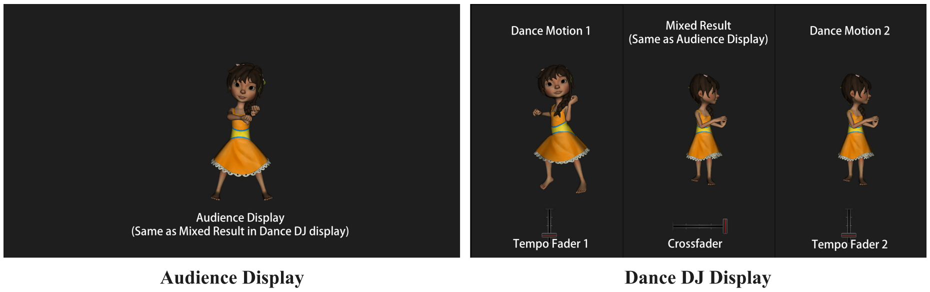
A typical DJ interface for music controls is shown in Figure 2. This interface has many buttons and sliders to control music. The left and right part that have similar buttons and sliders are for controlling two pieces of music respectively. The play button is for playing/pausing the music. The tempo faders adjust the speed of the music tempo. The turntables adjust playback speed to match tempos or beats. In the middle, the cross fader blends the two pieces of music assigned to the left and right parts. The sync buttons are used for automatic music tempo tuning based on the other assigned music.

To author the character's dance motion, we map the different dance motion control functions onto the DJ controller as shown in Table 1. The principle of such a mapping design is to conserve the meaning of the buttons such that DJs can transfer their skills from music controls to dance controls. The tempo faders are assigned with a motion speed control function. The play buttons start and pause a dance motion. The turntables are used to playback sequences of

Table 1. Mapping Function between DJ and Dance Parameters

DJ-Interface	Dance control parameters
Tempo fader	Motion speed
Cross fader	Mixing dance motion
Pinch fader	Sound effect
Turntable	Moving in few sequence
PLAY	Start/Stop dance animation
SYNC	Synchronization between music and dance
CUE	Visualization of the transition point

dance motion. The cross fader interpolates two different dance motions. The sync buttons are used for automatic tempo tuning of the selected dance motion based on the received music tempo. We use the cue button for the visualization function as described in Section 4 to find a smooth transition point between different dance motions.



Audience Display

Dance DJ Display

Fig. 3. DanceDJ's system screen in a prototype stage. (a) For audience, the center's character displayed for a user is only projected to the projection screen. (b) For a user, the system displays three characters, which left and right are assigned different motions respectively, and center is a result interpolated dance motions between left and right by using cross fader.

In Figure 3, we show a simple DanceDJ's system screen. From the user's perspective, the screen has three columns displaying the same character; the left and right columns display two different dance motions, and the center column displays the result that is interpolated from the left and right dance motions dependent on the cross fader's value. The audience can only see the resultant character in the center column.

Our system employs a data-driven approach to synthesize a new dance motion. We constructed a dance database that consisted of 16 dance motions from a Japanese video website *niconico*. In our study, this number of dance motions were sufficient for our experiments and a live performance for about half an hour. The dance motions were created by various amateur users, and the duration was about 3 minutes each. All dance motions are retargetted to the same structure with the same number of joints, which facilitates efficient motion interpolation. In our experiment, we used motion data with 70 joints included finger joints. Joints are represented using quaternions and we use *Spherical Linear Interpolation (SLERP)* for interpolating the joints between two dance motions.

It is a challenge to mix motions seamlessly and avoid sudden jumps of postures. While a DJ is able to mix different music by ad-lib as they have memorized the music, requiring the DJ to memorize all the dance choreography for dance mixing would be difficult. We address the problem by proposing a technical solution and by providing visual guidance. For the technical solution, we propose a novel transition function to evaluate a connectivity between two different dance motions based on the motion tempo, postures and the original music tempo assigned to each dance motion, which is detailed in Section 4. As the visual guidance, we visualize a result of the transition function to users on the system screen in real-time, which is described in Section 5.

4 A Transition Function for Dance Motions

We describe a transition function to evaluate how well a frame in a dance can seamlessly transition to a frame in another dance. This function supports a user to narrow down the transition frame candidates of a selected dance motion. It consists of two terms including (1) the beats of both music and motion $E_{beat}(i,j)$, and (2) the similarity of human poses $E_{pose}(i,j)$:

$$E_{trans}(i,j) = w_1 E_{beat}(i,j) + w_2 E_{pose}(i,j), \tag{1}$$

where i and j are the beat indexes of two different dance motions, w_1 and w_2 are weights from 0.0 to 1.0 for the evaluation functions $E_{beat}(i,j)$ and $E_{pose}(i,j)$ respectively. These weight parameters are controlled by the user using the pinch fader of the DJ interface. $E_{beat}(i,j)$ and $E_{pose}(i,j)$ are described in the next two subsections.

4.1 Beat Matching between Music and Motion

Considering that dance motion represents music beats as physical expression, the dance motion beat is related to joint angular velocity or angular moment. Therefore, we first compute the *Weight Effort* using a sum of angular velocity for each dance motion database. We then define motion beat from the minimum value in each a window range [21]. Since each dance motion comes with a corresponding piece of music, we obtain the window range from the beat of the music.

We use the *Songle* API to analyze the beat of the corresponding music. The beat information has a set of both time position and beat count, and we compute the beat per minutes (BPM) by calculating the average beat over time.

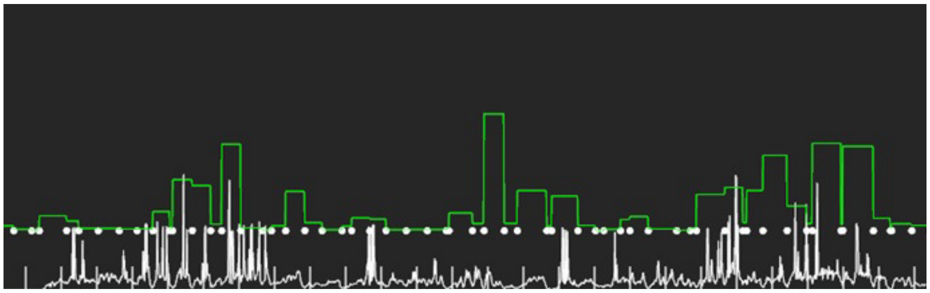


Fig. 4. An analyzed result for arbitrary dance motion. The wave information is *Weight Effort (WE)* considered the sum of angular velocity for the all joints at each frame. Dot circles represents dance motion beats calculated from the *WE* value

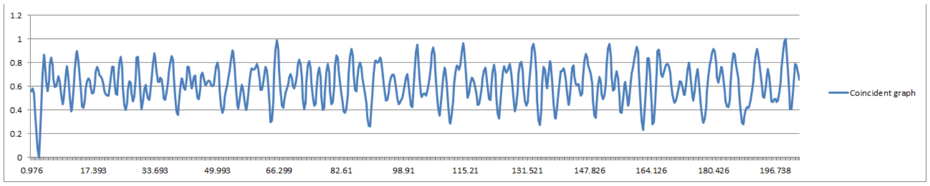


Fig. 5. The beat evaluation result (E_{beat}) of transition function applied for arbitrary music and the dance motion created by artists. When the music beat and the motion beat are completely matching, the result becomes a constant sine curve.

From the results as shown in Figure 4, we found that the dance motion beat does not always match the music beat because the dance motion beat is not at a constant interval. We expect that a smooth dance transition should happen at the frame when the beat of both music and motion matches. Therefore we design the beat evaluation function $E_{beat}(i, j)$ to evaluate a beat matching rate between the motion and the music. To evaluate a beat's coincidence factor of motion tempo with music tempo, we approximate the discretized motion tempo into a continuous function using a Gaussian distribution. The range of those values are from 0.0 to 1.0. We set 0.1 as the *sigma* value used in Gaussian distribution. The result is shown in Figure 5.

4.2 Posture Similarity

To seamlessly connect the different dance motions, we compute the similarity of the posture based on the sum of root mean square distance in joint positions for all joints [14]. Considering two dance motions, the frame-wise pose similarity can be calculated and represented using a similarity matrix as shown in Figure 6. The X-axis represents the frame number of a dance motion, and the Y-axis

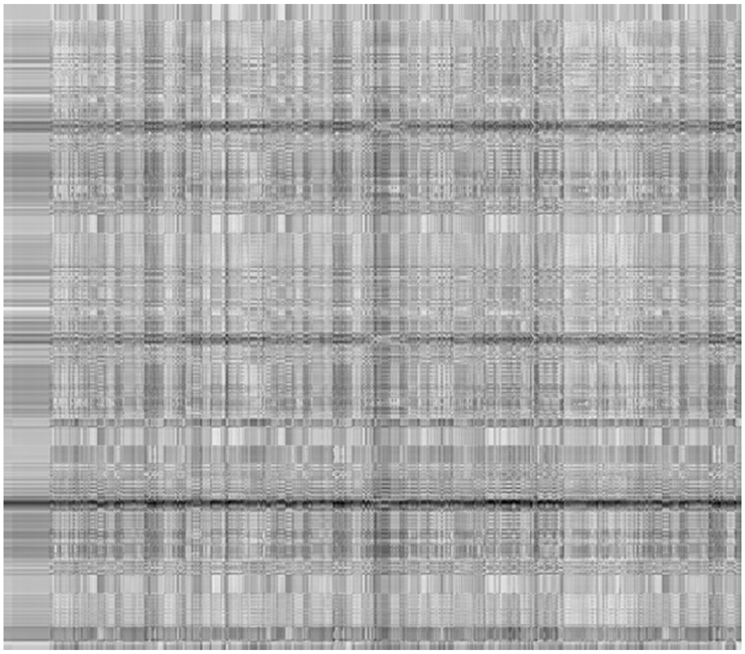


Fig. 6. A result of *Motion Graph* applied for two different dance motions; The width and height of the image corresponds to the number of frames for each two dance motions. The corresponded frame colored with white represents high similarity of two postures.

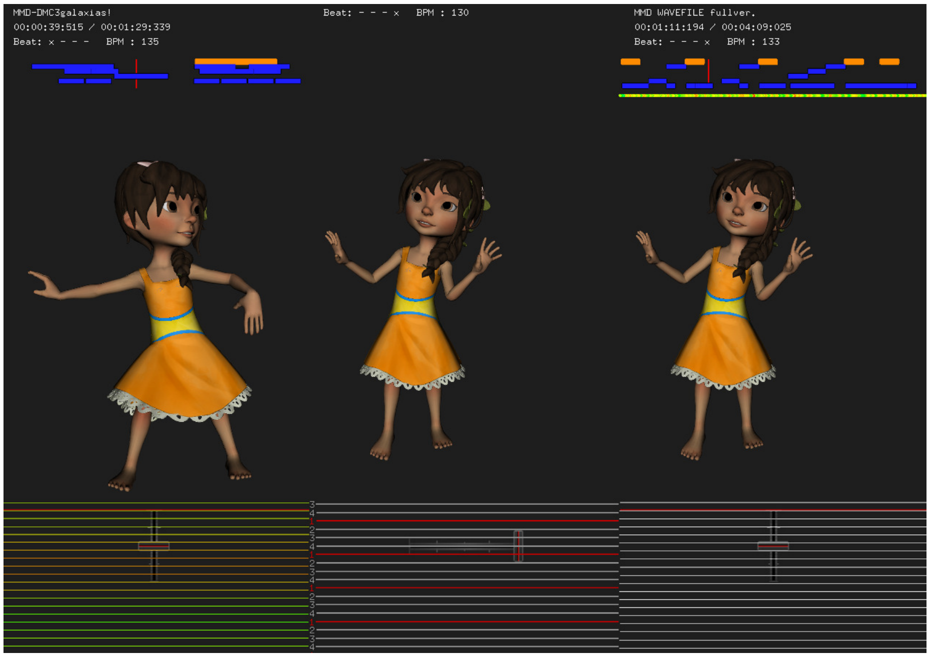
represents that of another dance. A darker pixel indicates a high similarity at the corresponding frame pair.

We conduct the similarity calculation for all dance motion combinations as an pre-process to reduce on-line computation time. During run-time, given two poses from two dance motions, the calculated value is retrieved as the result of the posture similarity function $E_{pose}(i, j)$.

5 Visual Guidance for Motion Transition

In this section, we describe how to visualize the dance motion's information on the system screen to assist a user in transitioning smoothly between two different dance motions.

An example of the graphical interface is shown in Figure 7. Our system has three partitions on the system screen; the left and right columns are for editing, and the center is for visualizing the mixing result. If a user sets the cross fader to the right side, the user can select the next dance motion and edit it on the left side.



MMD-DMC3galaxias!
00:00:39:515 / 00:01:29:339
Beat: x - - -   BPM : 135

Beat: - - - x   BPM : 130

MMD WAVEFILE fullver.
00:01:11:194 / 00:04:09:025
Beat: - - - x   BPM : 133

Fig. 7. Final system screen; we show the beat information below. In the upper right hand corner visual guidance based on the transition function for seamless transition in shown.

In particular, tn the top left and right hand corners of Figure 7, we display basic information including the input dance motion, the music title, duration, beat, BPM and structure. The lines at the bottom left and right hand sides show the music beat of the dance motion, which scrolls upwards during the playback of a dance motion. The corresponding color indicates the quality of transitions calculated from Equation 1. At the bottom of the center column, the colored lines represent music beat of four counts received from the DJ through *Ableton Link*. When the user pushes the Sync button on the DJ interface, the interval width of the music beat bars are fitted to the width of the beat received from the DJ.

5.1 Visualization of Transition Frames

Here, we describe a visual process for assisting the user in transitioning between different dance sequences.

When the user searches for the next transition point by pushing the cue button, the system uses Equation 1 to evaluate a smooth connection. A time-line that visualize the results of the transition function is shown below both the music structure. It indicate time-line information that takes the global similarity (based

Table 2. Questions for audiences; (7-points Likert scale, 7: most agree; 1: least agree)

No.	Question
Q1.	Were you satisfied with the animation result?
Q2.	Is the connection between different dance sequences natural?
Q3.	Did you feel that the animation result matched both the dance and music?

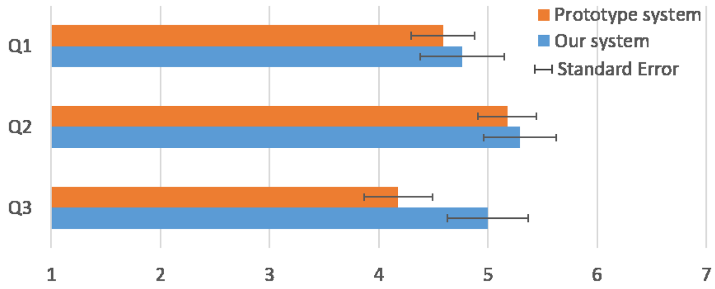


Fig. 8. Result of the user evaluation from audience perspective.

on music and motion features such as tempo and pose) with the other candidate dance motion into account. We use red to represent large values and blue to represent small values. Similarly, the color of the time-lines below the dancing character is updated using the same color scheme. It represents on the local similarity between the current selected dance frame and the other candidate's motion in the neighbor frame range.

6 User Study

To evaluate the effectiveness of our system, we conducted user studies from the perspectives of both audiences and users. There were 17 people (15 men and 2 women) and 12 people (11 men and 1 woman) for the user and audience perspective studies respectively. Their ages ranged from 21 to 30 years old. While no one had DJ experience, 25 percent of the subjects had dance experience and stage performance.

6.1 Audience Perspective

In the study of audience perspective, we first showed two dance animations using our system synchronized with music played by a DJ for a few minutes. The first animation was controlled by an experienced user without the transition function nor automatic beat synchronization. We call such a system the prototype system.

Table 3. Questions for users; Q1-Q4 (7-points Likert scale, 7: most agree; 1: least agree), Q5 (open ended)

No.	Question
Q1.	Could you naturally connect dance motion sequences?
Q2.	Could you match both dance and music?
Q3.	Were you satisfied with the dance animation you controlled.
Q4.	Did you feel that the mapping relationship between each button and motion function was adaptable?
Q5.	How long did it take you to control the dance motion satisfactorily?

The second animation was controlled by the same user using the full range of functions that we proposed. After showing both results, we asked subjects three questions for each synthesized animation shown in Table 2 to evaluate the visual effectiveness of the dance motion synthesized using the *transition function*. The results in Figure 8 show that our system consistently out-performs the prototype system in all three questions. It demonstrates that our system helps to improve the quality of dance mixing. We used a two-tailed Wilcoxon signed-rank test and found no significant differences in the results between the two systems (p-values were more than 0.05).

6.2 User Perspective

In the study of the user's perspective, we first explain the usage of the prototype system to the subject for 5 minutes. The subject then used the system while listening to a DJ playing music for 10 minutes. Next, we explained our full system consisting the visualized transition function and automatic beat synchronized function for 5 minutes. The subject then used it for 10 minutes.

We finally asked each subject to evaluate the systems with five questions shown in Table 3. Figure 9 shows the averaged results of the user study. In Q1, Q2 and Q3, our system scores significantly higher than the prototype system. A two-tailed Wilcoxon signed-rank test was conducted to demonstrate that the difference between the average score was statistically significant. In Q4, most subjects suggested that our system was more adaptable in mapping transition between dance motion than the prototype system, as shown in Figure 10. In Q5, the transition function with the visual guidance allows subjects to intuitively control without taking much time, as shown in Figure 11.

The results of user evaluation show that our method was more effective in transitioning seamlessly from both the audience and user's perspective.

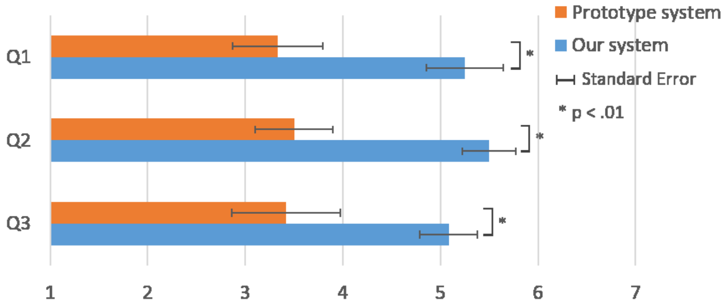


Fig. 9. Result of the user evaluation from user perspective.

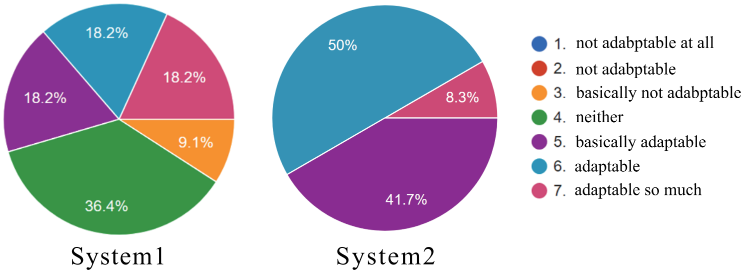


Fig. 10. Question 4

6.3 Other Feedback

As a separate user study, we also conducted a live performance with a DJ in a music club. The synthesized character's dance motion was projected to a large 2D screen using a projector. There were about 20-30 audience. There were positive reaction from the audience when the character was synchronized with the DJ's music. During our live performance we met a opportunity to collaborate with VJ).We mixed two video channel half and half, and the VJ made the visual effects such as the dance floor for our dancing character by ad-lib. This is shown in Figure 12.

7 Discussions

Experimental feedback : As an evaluation, we conducted two experiments to evaluate both audience and user perspective. Since all the subjects were asked to evaluate the prototype prior to evaluating our Dance DJ, comparison might not be fairly evaluated. To counterbalance the evaluation more, we conducted an additional experiment in reverse order to evaluate detailed advantages of our

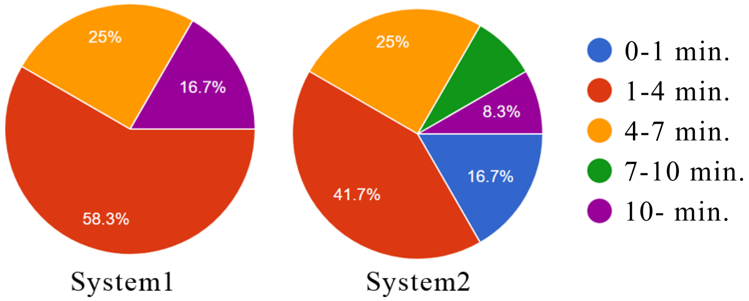


Fig. 11. Question 5

Fig. 12. The scene of live performance. Left and right persons are DanceDJ player and DJ player. The middle person is next VJ player. Incidentally, DanceDJ player and a VJ player started a novel collaboration.

system. From our experiments, we found that DanceDJ has some rooms of improvement for audiences and users. For example, because our system can assist user to seamlessly connect different dance motion naturally, sometimes many audiences do not notice the transition's timing. This raises the problem that the system may not encourage dramatic effects at the transition like what DJs do between drastically different songs. As our future work, we will design diverse connecting function, such as for a mash-up affection, which provide user intuitive control to create different varieties of dance motion simply by changing the function. Some future directions are listed below:

Emotional Enrichment: We wish to visualize the emotional features of music in real-time. A character's facial and body motion are synchronized with music emotional features perceived by the audiences. We then integrate existing techniques such as automatic facial motion synthesis [1] and motion filter [24] using the pinch fader to enrich quality of the animation.

Database Limitation: For all the experiments, we used a motion database created by armature artists. Hence, the quality and the number of the motion database are limited. In the future, we will collect more dance motions and corresponded music data-set from the video shared in different video hosting services.

Helping Users with Disabilities: Since it only requires a few simple control to create the dance motion, we believe that the system could help those who cannot dance due to physical disabilities for enjoying dancing. We plan to evaluate how to better serve such a user group by providing a system to generate virtual dance characters.

Artists Collaboration: We believe that the users of our system, DanceDJs, have a possibility to be a type of major artists in the field, such as DJs or VJs. Since the collaboration with DJs has proven to be successful, the collaborations with VJs will be a very interesting direction to move forward for exploring future possibilities.

8 Conclusion

We have presented DanceDJ, a novel DJ-like controller for creating dance animation in real-time. We have presented the transition function and visual guidance for user control. The transition function considered both synchronization between the music and human motion beats, and posture similarity with next dance's posture. By visualizing the transition function's results, this system allows the user to intuitively and effectively transition into the next dance motion. We evaluated the effectiveness from audience and user perspectives by conducting user evaluation.

In our future work, we will add a motion effect function for changing the dynamics of dance motions with the pinch fader. By collaborating with Video Jockeys (VJs), we will represent the dancing character and the background environment more attractively. Furthermore, our system will adopt a motion exploring function to find arbitrary dance motions from dance motion database as soon as possible as an on-line process. We believe DanceDJ has a good potential to conduct a new style of live performance in collaboration with DJ, VJ and real dancers on a stage.

Acknowledgement

This work was supported in part by JST ACCEL Grant Number JPMJAC1602, Japan. It was also supported by the Engineering and Physical Sciences Research Council (EPSRC) (Ref: EP/M002632/1) and the Royal Society (Ref: IE160609).

References

[1] Asahina, W., Okada, N., Iwamoto, N., Masuda, T., Fukusato, T., Morishima, S.: Automatic facial animation generation system of dancing characters considering emotion in dance and music. In: SIGGRAPH Asia 2015 Posters. pp. 11:1–11:1. SA '15, ACM, New York, NY, USA (2015), http://doi.acm.org/10.1145/2820926.2820935

[2] Choi, B., i Ribera, R.B., Lewis, J.P., Seol, Y., Hong, S., Eom, H., Jung, S., Noh, J.: Sketchimo: Sketch-based motion editing for articulated characters. ACM Trans. Graph. 35(4), 146:1–146:12 (jul 2016), http://doi.acm.org/10.1145/2897824.2925970

[3] Choi, M.G., Yang, K., Igarashi, T., Mitani, J., Lee, J.: Retrieval and visualization of human motion data via stick figures. Comput. Graph. Forum 31(7pt1), 2057–2065 (sep 2012), http://dx.doi.org/10.1111/j.1467-8659.2012.03198.x

[4] Dontcheva, M., Yngve, G., Popović, Z.: Layered acting for character animation. In: ACM SIGGRAPH 2003 Papers. pp. 409–416. SIGGRAPH '03, ACM, New York, NY, USA (2003), http://doi.acm.org/10.1145/1201775.882285

[5] Fender, A., Müller, J., Lindlbauer, D.: Creature teacher: A performance-based animation system for creating cyclic movements. In: Proceedings of the 3rd ACM Symposium on Spatial User Interaction. pp. 113–122. SUI '15, ACM, New York, NY, USA (2015), http://doi.acm.org/10.1145/2788940.2788944

[6] Glauser, O., Ma, W.C., Panozzo, D., Jacobson, A., Hilliges, O., Sorkine-Hornung, O.: Rig animation with a tangible and modular input device. ACM Trans. Graph. 35(4), 144:1–144:11 (jul 2016), http://doi.acm.org/10.1145/2897824.2925909

[7] Groth, P., Shamma, D.A.: Spinning data: Remixing live data like a music dj. In: CHI '13 Extended Abstracts on Human Factors in Computing Systems. pp. 3063–3066. CHI EA '13, ACM, New York, NY, USA (2013), http://doi.acm.org/10.1145/2468356.2479611

[8] Guay, M., Cani, M.P., Ronfard, R.: The line of action: An intuitive interface for expressive character posing. ACM Trans. Graph. 32(6), 205:1–205:8 (nov 2013), http://doi.acm.org/10.1145/2508363.2508397

[9] Hahn, F., Mutzel, F., Coros, S., Thomaszewski, B., Nitti, M., Gross, M., Sumner, R.W.: Sketch abstractions for character posing. In: Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation. pp. 185–191. SCA '15, ACM, New York, NY, USA (2015), http://doi.acm.org/10.1145/2786784.2786785

[10] Held, R., Gupta, A., Curless, B., Agrawala, M.: 3d puppetry: A kinect-based interface for 3d animation. In: Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology. pp. 423–434. UIST '12, ACM, New York, NY, USA (2012), http://doi.acm.org/10.1145/2380116.2380170

[11] Ishigaki, S., White, T., Zordan, V.B., Liu, C.K.: Performance-based control interface for character animation. ACM Transactions on Graphics (SIGGRAPH) 28(3) (2009)

[12] Jacobson, A., Panozzo, D., Glauser, O., Pradalier, C., Hilliges, O., Sorkine-Hornung, O.: Tangible and modular input device for character articulation. ACM Trans. Graph. 33(4), 82:1–82:12 (jul 2014), http://doi.acm.org/10.1145/2601097.2601112

[13] Jin, M., Gopstein, D., Gingold, Y., Nealen, A.: Animesh: Interleaved animation, modeling, and editing. ACM Trans. Graph. 34(6), 207:1–207:8 (oct 2015), http://doi.acm.org/10.1145/2816795.2818114

[14] Kovar, L., Gleicher, M., Pighin, F.: Motion graphs. ACM Trans. Graph. 21(3), 473–482 (jul 2002), http://doi.acm.org/10.1145/566654.566605

[15] Lee, J., Chai, J., Reitsma, P.S.A., Hodgins, J.K., Pollard, N.S.: Interactive control of avatars animated with human motion data. ACM Trans. Graph. 21(3), 491–500 (jul 2002), http://doi.acm.org/10.1145/566654.566607

[16] Liu, Z., Huang, J., Bu, S., Han, J., Tang, X., Li, X.: Template deformation-based 3-d reconstruction of full human body scans from low-cost depth cameras. IEEE Transactions on Cybernetics 47(3), 695–708 (March 2017)

[17] Liu, Z., Zhou, L., Leung, H., Shum, H.P.H.: Kinect posture reconstruction based on a local mixture of gaussian process models. IEEE Transactions on Visualization and Computer Graphics 22(11), 2437–2450 (Nov 2016)

[18] Norman, A., Amatriain, X.: Data jockey, a tool for meta-data enhanced digital djing and active listening. In: ICMC. Michigan Publishing (2007)

[19] Ragnhild, M.M., Mckelvin, M., Nest, R., Valdez, L., ping Yee, K., Back, M., Harrison, S.: Seismospin: A physical instrument for digital data. In: In CHI f03: CHI f03 extended abstracts on Human factors in computing systems. pp. 832–833. ACM Press (2003)

[20] Shiratori, T., Hodgins, J.K.: Accelerometer-based user interfaces for the control of a physically simulated character. ACM Trans. Graph. 27(5), 123:1–123:9 (dec 2008), http://doi.acm.org/10.1145/1409060.1409076

[21] Shiratori, T., Nakazawa, A., Ikeuchi, K.: Dancing-to-music character animation. Comput. Graph. Forum 25(3), 449–458 (2006), http://dx.doi.org/10.1111/j.1467-8659.2006.00964.x

[22] Shirokura, T., Sakamoto, D., Sugiura, Y., Ono, T., Inami, M., Igarashi, T.: Robojockey: Real-time, simultaneous, and continuous creation of robot actions for everyone. In: Proceedings of the 7th International Conference on Advances in Computer Entertainment Technology. pp. 53–56. ACE '10, ACM, New York, NY, USA (2010), http://doi.acm.org/10.1145/1971630.1971646

[23] Shum, H.P.H., Ho, E.S.L., Jiang, Y., Takagi, S.: Real-time posture reconstruction for microsoft kinect. IEEE Transactions on Cybernetics 43(5), 1357–1369 (2013)

[24] Wang, J., Drucker, S.M., Agrawala, M., Cohen, M.F.: The cartoon animation filter. ACM Trans. Graph. 25(3), 1169–1173 (Jul 2006), http://doi.acm.org/10.1145/1141911.1142010

[25] Yazaki, Y., Soga, A., Umino, B., Hirayama, M.: Automatic composition by body-part motion synthesis for supporting dance creation. In: 2015 International Conference on Cyberworlds (CW). pp. 200–203 (Oct 2015)

[26] Yoshizaki, W., Sugiura, Y., Chiou, A.C., Hashimoto, S., Inami, M., Igarashi, T., Akazawa, Y., Kawachi, K., Kagami, S., Mochimaru, M.: An actuated physical puppet as an input device for controlling a digital manikin. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 637–646. CHI '11, ACM, New York, NY, USA (2011), http://doi.acm.org/10.1145/1978942.1979034

[27] Zhai, S., Milgram, P.: Quantifying coordination in multiple dof movement and its application to evaluating 6 dof input devices. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 320–327. CHI '98, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA (1998), http://dx.doi.org/10.1145/274644.274689

[28] Zhang, P., Siu, K., Zhang, J., Liu, C.K., Chai, J.: Leveraging depth cameras and wearable pressure sensors for full-body kinematics and dynamics capture. ACM Trans. Graph. 33(6), 221:1–221:14 (Nov 2014), http://doi.acm.org/10.1145/2661229.2661286