Query-by-Dancing: A Dance Music Retrieval System Based on Body-Motion Similarity

Shuhei Tsuchida(✉), Satoru Fukayama(✉), and Masataka Goto(✉)

National Institute of Advanced Industrial Science and Technology (AIST),
Central 2, 1-1-1 Umezono, Tsukuba, Ibaraki, Japan
{s-tsuchida,s.fukayama,m.goto}@aist.go.jp

Abstract. This paper presents Query-by-Dancing, a dance music retrieval system that enables a user to retrieve music using dance motions. When dancers search for music to play when dancing, they sometimes find it by referring to online dance videos in which the dancers use motions similar to their own dance. However, previous music retrieval systems could not support retrieval specialized for dancing because they do not accept dance motions as a query. Therefore, we developed our Query-by-Dancing system, which uses a video of a dancer (user) as the input query to search a database of dance videos. The query video is recorded using an ordinary RGB camera that does not obtain depth information, like a smartphone camera. The poses and motions in the query are then analyzed and used to retrieve dance videos with similar poses and motions. The system then enables the user to browse the music attached to the videos it retrieves so that the user can find a piece that is appropriate for their dancing. An interesting problem here is that a simple search for the most similar videos based on dance motions sometimes includes results that do not match the intended dance genre. We solved this by using a novel measure similar to tf-idf to weight the importance of dance motions when retrieving videos. We conducted comparative experiments with 4 dance genres and confirmed that the system gained an average of 3 or more evaluation points for 3 dance genres (waack, pop, break) and that our proposed method was able to deal with different dance genres.

Keywords: Dance · Music · Video · Retrieval system · Body-motion

1 Introduction

Dancers often dance to music. They choose a dancing style that can match the genre or style of a musical piece, synchronize their movements with musical beats and downbeats, and change their movements to follow musical changes. When musical pieces are played on a dance performance stage, for example, the dancers just have to dance to match the piece being performed. On the

I. Kompatsiaris et al. (Eds.): MMM 2019, LNCS 11295, pp. 251–263, 2019.
https://doi.org/10.1007/978-3-030-05710-7_21

other hand, when dancers can select musical pieces for their dance performances, practices, or personal enjoyment, they spend a lot of time finding musical pieces appropriate for their intended performance. This is because selecting pieces of music is important for achieving successful dance performances and enjoying dancing. When searching for music to play while dancing, dancers sometimes find it by referring to online dance videos in which the dancers use motions similar to their own dance. They may also refer to dance events or showcases of their favorite dancers or dance groups for music. Since there has been no systematic support for finding certain kinds of dance music, such activities have been time consuming and difficult. Although many music retrieval systems have been proposed [2,7,9], none have focused on retrieving dance music.

Therefore, we developed a dance music retrieval system called Query-by-Dancing that enables a dancer (user) to use his/her dance motions to retrieve music. To find music for the dancer to dance to, our system first finds a dance music video that contains dancing similar to the motions of the user's dance. Our system can retrieve dance videos that include motions similar to an input query of a short video capturing dancing body motions. The musical pieces in the videos should be appropriate for the user to dance to.

Our Query-by-Dancing system does not need an expensive high-performance motion capture system or a camera that obtains depth information. It only needs a simple RGB camera like those installed in smartphones. We implemented a system that analyzes input query videos and a database of dance videos by using the OpenPose library by Cao et al. [1] to estimate body motions.

2 Related Work

A number of music retrieval and recommendation systems have been proposed, but none have allowed a dancer to search for music using dance motions. Our system, Query-by-Dancing, is equipped with a novel function based on the similarities between dance motions that enables the user to input their own dance video as a retrieval query. We surveyed studies on music retrieval and recommendation systems using various queries.

Ghias et al. [4] proposed a query-by-humming system that uses humming as a query. They claim that an effective and natural way of searching a musical audio database is by humming the song. Chen et al. [3] proposed a system for retrieving songs from music databases using rhythm. They use strings of notes as music information, and the database returns all songs containing patterns similar to the query. Jang et al. [5] proposed a query-by-tapping system. The system allows the user to search a music database by tapping on a microphone to input the duration of the first several notes of the query song. Maezawa et al. [6] proposed a query-by-conducting system. In this system, the interface allows a user to conduct during the playback of a piece, and the interface dynamically switches the playback to a musical piece that is similar to the user's conducting.

Some systems retrieve musical pieces by using a musical context, such as the artist's cultural or political background, collaborative semantic labels, and album cover artwork [10]. Turnbull et al. [11] presented a query-by-text system that can use a text-based query to retrieve relevant tracks from a database of unlabeled audio content. This system can also annotate novel audio tracks with meaningful words.

As described above, several retrieval methods using various queries have been proposed. However, to the best of our knowledge, this is the first study that has used dance motions as a query for retrieving music. Our system focuses on dance motions and acquires candidate musical pieces from a dance video database.

3 Dance Music Retrieval System

The system overview is shown in Fig. 1. Our system can be divided into two stages: pre-processing and similarity calculation. These two main stages are described below.

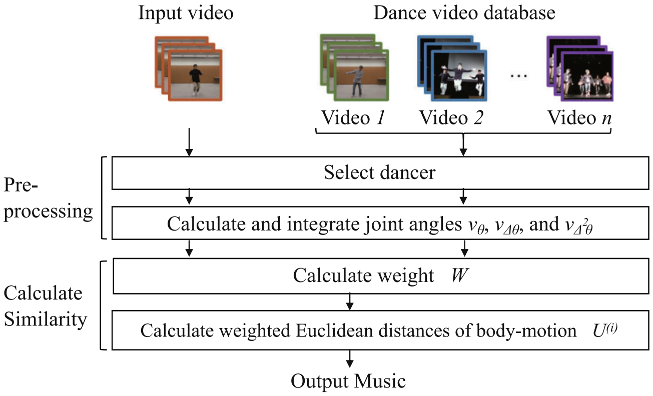
Input video
Dance video database
Video 1
Video 2
...
Video n
Pre-processing
Select dancer
Calculate and integrate joint angles v_θ, $v_{\Delta\theta}$, and $v_{\Delta^2\theta}$
Calculate Similarity
Calculate weight W
Calculate weighted Euclidean distances of body-motion $U^{(i)}$
Output Music

Fig. 1. System overview.

3.1 Pre-processing

Detect a Dancer. In this step, the system first estimates the person's skeleton information in all video frames using the OpenPose library [1]. Dance videos sometimes include frames of multiple people dancing or the OpenPose library sometimes detects skeleton information incorrectly on a frame that has no person. Therefore, the system selects the skeleton representing the main dancer by analyzing all of the skeletons detected in each frame. First, the area (A_o) occupied by each skeleton detected is computed by multiplying the width by the height of the area. The width is defined as the difference between the maximum and minimum values along the x-axis direction of the detected skeleton(s). The height is defined as the difference between the maximum and minimum values in the y-axis direction of the skeleton(s). Then, the position (P_d) of a dancer in a video is obtained by averaging the skeleton positions in all of the frames. The distance (D_c) from the center ($P_\mathrm{c}(X_\mathrm{mean}, Y_\mathrm{mean})$) of the entire frame image to P_d is computed for each dancer. Assuming the main dancer is located in the center, the skeleton that maximizes $R = \frac{A_\mathrm{o}}{D_\mathrm{c}}$ is selected as the main dancer.

Feature Extraction. To calculate the similarity between dance motions in the query video and each dance video in the database, we extracted the following motion features from the skeleton of the main dancer. Since both poses and motions are important elements that characterize dancing, the system first represents the poses by calculating 17 joint angles from the skeleton per frame. The angle is calculated clockwise from the upper side vertically across the image as zero. The joint angles are broken into the two dimensions θ_x and θ_y by calculating sine and cosine (as shown in Fig. 2), and we denote a 34-dimensional feature vector of angles at n-th frame of i-th video by $v_\theta^{(i)}(n)(1 \leq n \leq N^{(i)} and\, 1 \leq i \leq I)$, where N is the number of frames in the i-th video and I is the number of videos in the database. Furthermore, the angle where the skeleton was not detected is expressed as zero.

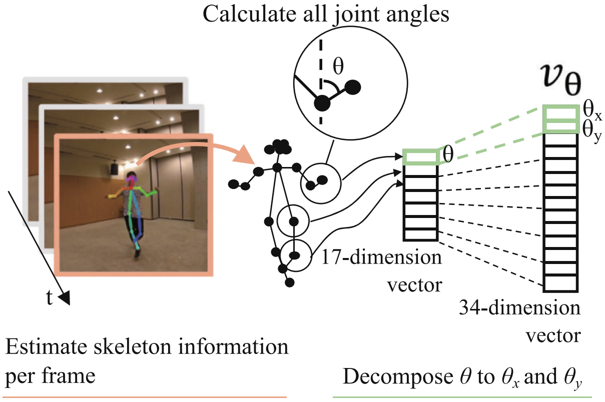
Calculate all joint angles
θ
v_θ
θx
θy
θ
t
17-dimension vector
34-dimension vector
Estimate skeleton information per frame
Decompose θ to θ_x and θ_y

Fig. 2. All the joint angles are broken down per frame, and creating a 34-dimensional vector.

The system then represents the body motions by calculating the speed and acceleration of the change in joint angles between frames. It calculates $v_{\Delta\theta}^{(i)}(n)$ and $v_{\Delta^2\theta}^{(i)}(n)$ as follows:

$$v_{\Delta\theta}^{(i)}(n) = \mathrm{abs}(v_\theta^{(i)}(n) - v_\theta^{(i)}(n-1)) \tag{1}$$

$$v_{\Delta^2\theta}^{(i)}(n) = \mathrm{abs}(v_{\Delta\theta}^{(i)}(n) - v_{\Delta\theta}^{(i)}(n-1)) \tag{2}$$

where $\mathrm{abs}(x)$ denotes a vector containing the absolute value of each element of x. We concatenated the above 3 feature vectors, $v_\theta^{(i)}(n)$, $v_{\Delta\theta}^{(i)}(n)$, and $v_{\Delta^2\theta}^{(i)}(n)$, into one 102-dimensional vector $v^{(i)}(n)$.

3.2 Similarity Calculation

The system calculates the Euclidean distances $d(v^{\text{in}}(n), v^{(i)}(m))$ between all frames ($1 \leq n \leq N^{\text{in}}$) of an input video (in) and all frames ($1 \leq m \leq N^{(i)}$) of a video in the video database ($1 \leq i \leq I$), where $d(x, y)$ denotes an Euclidean distance between x and y, as shown in Fig. 3. The system computes these Euclidean distances in all frame combinations and divides them by the total number of combinations ($N^{\text{in}}N^{(i)}$). They are denoted by following formula:

$$R^{(i)} = \frac{1}{N^{\text{in}}N^{(i)}} \sum^{N^{\text{in}}} \sum^{N^{(i)}} d(v^{\text{in}}(n), v^{(i)}(m)). \tag{3}$$

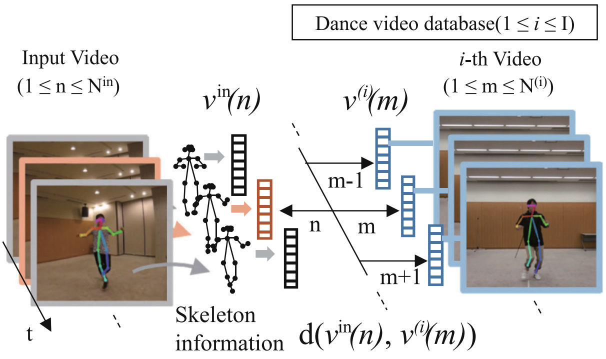
Dance video database($1 \leq i \leq$ I)
Input Video
($1 \leq$ n $\leq$ N$^{\text{in}}$)
$v^{\text{in}}(n)$
$v^{(i)}(m)$
i-th Video
($1 \leq$ m $\leq$ N$^{(i)}$)
m-1
n
m
m+1
t
Skeleton information
d($v^{\text{in}}(n)$, $v^{(i)}(m)$)

Fig. 3. The system computes the Euclidean distances in all frame combinations and divides them by the total number of combinations.

Now the system simply finds the most similar videos by using $R^{(i)}$ as the similarity of the dance motions per video, which leads to results that do not match the intended dance genre. We solved this problem by using a novel measure similar to tf-idf to weight the importance of dance motions when retrieving videos. The weight representing the importance of dance motions is calculated as follows:

$$W(n) = \frac{\frac{1}{N^{(i)}} \sum^{N^{(i)}} d(v^{\text{in}}(n), v^{(i)}(m))}{\max\limits_{i \in I}\{\frac{1}{N^{(i)}} \sum^{N^{(i)}} d(v^{\text{in}}(n), v^{(i)}(m)\}}. \tag{4}$$

where $\max(x)$ denotes a maximum value of x. To sharpen the weight gradient, the system calculates $W(n)$ to the 30-th power to obtain $W'(n)$. We determined the exponent 30 experimentally. Then, the system multiplied $W'(n)$ by all of the Euclidean distances. These distances are determined with the following formula:

$$U^{(i)} = \frac{\sum^{N^{\text{in}}} (W'(n) \sum^{N^{(i)}} d(v^{\text{in}}(n), v^{(i)}(m)))}{N^{\text{in}}N^{(i)}}. \tag{5}$$

The system finds the videos with the top k values among $U^{(i)}$ as the videos that contain dance motions similar to those in the input video. Finally, the system presents the candidate musical pieces from the dance videos searched.

4 Evaluation

We conducted two evaluation experiments to investigate whether the retrieval results are easy for dancers to use as dance music. In the first experiment, the system retrieved dance music based on whether the importance of each video was weighted or not. We adapted the calculated weights to the two methods; ADD method and DTW method, and retrieved dance music using each method. In the second experiment, the system retrieved dance music by using the dance videos of 4 dance genres as queries. This retrieval was done using the method that gave the best results in the first experiment.

4.1 Dance Video Database

We used 100 dance videos available on YouTube and Instagram. They were 25 videos per each dance genre we chose—break, hip-hop, waack, and pop—and their average duration was 82 s. The audio track of all these videos contained music that the dancers in the videos danced to.

4.2 Experiment I: Weighted or Unweighted

Experiment Conditions. We recruited 12 participants (4 males and 8 females) who were students belonging to a street dance club. All had between 1 to 15 years of dance experience (average = 8.5 years).

We compared 4 retrieval methods: ADD (unweighted), ADD (weighted), DTW (unweighted), and DTW (weighted). The ADD method, our proposed retrieval method using the Euclidean distance between frames, calculates one feature vector $v^{(i)}(n)$ of 102 dimensions by concatenating $v_{\theta}^{(i)}(n)$, $v_{\Delta\theta}^{(i)}(n)$, and $v_{\Delta^2\theta}^{(i)}(n)$ per frame. ADD (unweighted) does not use $W'(n)$, and it lists musical pieces in ascending order of $R^{(i)}$. ADD (weighted) uses $W'(n)$, and it lists musical pieces in ascending order of $U^{(i)}$.

The DTW method is a retrieval method that uses dynamic time warping, a sequence matching algorithm that considers longer-term similarity. This method creates a sequence $V_{\mathrm{dtw}}^{(i)}(n)(1 \leq n \leq N^{(i)} - 5 \,\text{and}\, 1 \leq i \leq I)$ for every 6 frames by sliding $v_{\theta}^{(i)}(n)$ one frame at a time. The system calculates the dynamic time warping $\mathrm{dtw}(v_{\mathrm{dtw}}^{\mathrm{in}}(n), V_{\mathrm{dtw}}^{(i)}(m))$ between all sequences $(1 \leq n \leq N^{\mathrm{in}} - 5)$ of an input video (in) and all sequences $(1 \leq m \leq N^{(i)} - 5)$ of a video in the video database $(1 \leq i \leq I)$, where $\mathrm{dtw}(x, y)$ is the Euclidean distance between x and y calculated by FastDTW [8]. Then, $R_{\mathrm{dtw}}^{(i)}$ and $W_{\mathrm{dtw}}(n)$ are calculated using an equation in which the $d(v^{\mathrm{in}}(n), v^{(i)}(m))$ in Eq. (3) are replaced with $\mathrm{dtw}(v_{\mathrm{dtw}}^{\mathrm{in}}(n), V_{\mathrm{dtw}}^{(i)}(m))$. To sharpen the weight gradient, the system calculates

$W_{\text{dtw}}(n)$ to the 40-th power to obtain $W'_{\text{dtw}}(n)$. We determined the exponent 40 experimentally. Then, the system multiplies $W'_{\text{dtw}}(n)$ by all the Euclidean distances and obtains $U^{(i)}_{\text{dtw}}$. DTW (unweighted) does not use $W'_{\text{dtw}}(n)$, and it lists musical pieces in ascending order of $R^{(i)}_{\text{dtw}}$. DTW (weighted) uses $W'_{\text{dtw}}(n)$ and lists musical pieces in ascending order of $U^{(i)}_{\text{dtw}}$.

We asked a waack dancer with 15 years of dance experience to participate in the experiment and shot about 11 s of her waack dancing. With that video as a query, we used each of the 4 methods to retrieve musical pieces. We denoted the top 5 music groups in the retrieval results obtained with each of the 4 methods as MG-A, MG-B, MG-C, and MG-D. Each music group had 5 musical pieces.

Procedure. At the beginning of the session, participants filled out a pre-study questionnaire about their dance experience. Then, we gave them a brief explanation of the experiment. After watching a query of the waack dancer's 11-s video without music, they were asked to listen to the 5 musical pieces in each music group and evaluate them on a 5-point Likert scale ranging from 1 for "do not agree" to 5 for "totally agree." They were given the music groups MG-A, MG-B, MG-C, and MG-D in random order. We gave them the evaluation item below.

Q1: Given the assumption that "someone" dances with the choreography shown in this video while listening to music, is each of these 5 musical pieces easy to dance to according to its atmosphere and the atmosphere of the choreography?

Finally, they filled out a questionnaire about the dance music retrieval.

We prepared a MacBook Pro (Retina display, 15-in., mid-2015) and used the QuickTime Player to play the musical pieces and the video. The query dance video was set to "repeat play" beforehand and the participant selected and played the 5 musical pieces arranged next to it. The participants wore earphones to listen to the music and could play and re-evaluate the musical pieces as many times as they wanted. The participants could take breaks freely during the experiment. The experiment took about 40 min.

Results and Discussion. The averaged Q1 scores for each retrieval method are shown in Fig. 4. The vertical axis indicates the average of the Q1 scores given by all of the participants, and the vertical bars indicate standard errors. The horizontal axis represents retrieval methods. The gray rectangles show the averaged evaluation scores for each of the retrieval ranks. Each green rectangle shows the average of all evaluation scores within the retrieval method. We assessed the difference between the average Q1 scores with ANOVA. There was a significant difference ($F_{(3,236)} = 4.21, p < .05$). We also assessed the difference with Fisher's Least Significant Difference (LSD) test and found significant differences ($p < .05$) between ADD (weighted) and the other 3 methods. Thus, ADD (weighted) was the suitable retrieval method of searching for musical pieces that dancers can easily dance to.

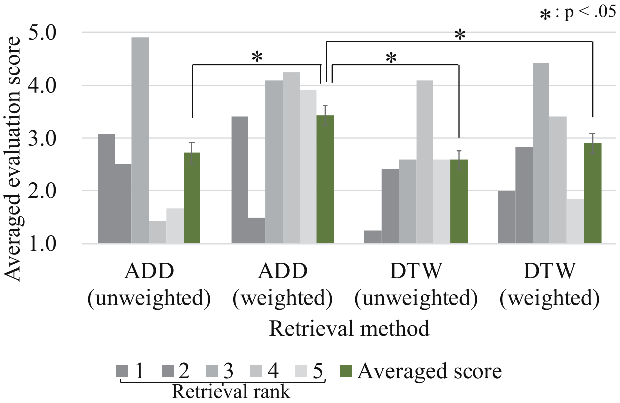

Fig. 4. The averaged Q1 scores for each retrieval method. The ADD (weighted) method was evaluated as significantly higher than the other 3 methods.

The retrieval results of ADD (weighted) had the same dance genres as the query more often than the other retrieval methods, which increased the evaluation score of ADD (weighted). The dance genres of each retrieval rank for each retrieval method are shown in Table 1. Focusing on the top 5 musical pieces in the retrieval results, we found that waack was 4 out of 5 musical pieces for ADD (weighted), which was the same dance genre as the query. For the other methods, 2 out of 5, 1 out of 5, and 3 out of 5 musical pieces were waack. The musical pieces used in videos of the same genre as the query's got higher evaluation scores.

Table 1. Top 5 retrieval results by retrieval methods. P in the table stands for the dance genre pop, and W stands for the dance genre waack.

Retrieval method	Dance genre	Retrieval rank				
		1	2	3	4	5
ADD (unweighted)	Waack	W	P	W	P	P
ADD (weighted)		W	P	W	W	W
DTW (unweighted)		P	P	W	P	P
DTW (weighted)		P	W	W	W	P

Next, we focused on the weights. Figure 4 shows that the scores of the weighted methods were higher than the unweighted methods', and that weighting is effective for retrieving dance music appropriate for dance motions. The calculated weight $W'(n)$ is shown in Fig. 5, where the vertical axis indicates the weight value and the horizontal axis represents the frame numbers in the video used as the query. The high-weight movements in the vicinity of frames 240 to 270 were movements such as the dancer swinging her arm above her head in long

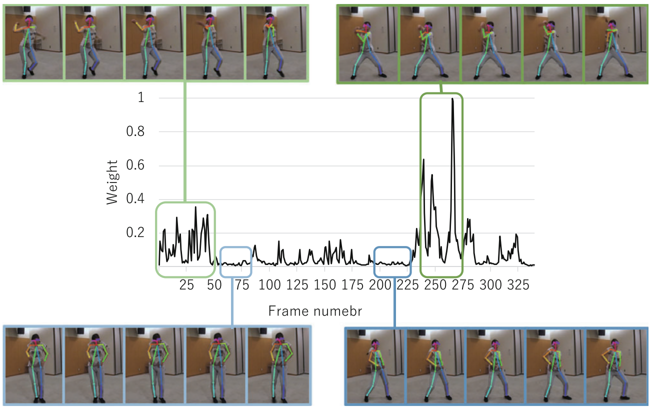

Fig. 5. Waack's characteristic movements were in the vicinity of a relatively high weight value. The movements common to other dance genres were in the vicinity of a relatively low weight value.

strides. Moreover, the movements in the vicinity of the first 50 frames with a relatively high weight value were movements such as the dancer swinging her arm to the left and right in long strokes. Swinging the arm in long strokes, a characteristic waack movement, had been highly weighted. On the other hand, the movements in the vicinity of frames 50 to 75 with a relatively low weight value were simple movements like moving backwards. Moreover, the movements in the vicinity of frames 200 to 225 with a relatively low weight value were movements such as the dancer shaking her waist to the left and right. These movements also occur in other dance genres. As the above shows, the system successfully weighted the movement particular to the dance motion in the query. In contrast, movements common to other dance genres were weighted low.

4.3 Experiment II: Retrieval Performance

Experiment Conditions. We recruited 12 participants (6 males and 6 females) who were students belonging to a street dance club. All participants had 1 to 15 years of dance experience (average = 5.9 years). We compared 4 dance genres: waack, break, pop, and hip-hop. We prepared the waack video used in the first experiment. The author who has 8 years of dance experience was in charge of a breakdancer, and we shot about 13 s of that author's breakdancing. To prepare other videos, we recruited two more dancers, a pop dancer and a hip-hop dancer. The pop dancer had 3 years of dance experience, and we shot about 16 s of his pop dance. The hip-hop dancer had 15 years of dance experience, and we shot about 16 s of her hip-hop dance. Using those videos as queries, we retrieved musical pieces by using the ADD (weighted) method. We denoted the top 5 music groups in the retrieval results obtained in each of the dance genres as DG-W, DG-B, DG-H, and DG-P. Each music group had 5 musical pieces.

Procedure. At the beginning of the session, participants filled out a pre-study questionnaire about their dance experience. Then, we briefly explained the experiment. After watching a query that was one of the randomly selected dance videos without music, they were asked to listen to the 5 musical pieces in each music group and evaluate them on a 5-point Likert scale ranging from 1 for "do not agree" to 5 for "totally agree." They were given the music groups DG-W, DG-B, DG-H, and DG-P according to the dance genre of the video. We gave Q1 as the evaluation item. Finally, they were orally interviewed.

We prepared a MacBook Pro (Retina display, 15-in., mid-2015) and used the QuickTime Player to play the musical pieces and the video. The query dance video was set to "repeat play" beforehand, and the participants selected and played the 5 musical pieces arranged next to it. The participants wore earphones to listen to the music, and they could play and re-evaluate the musical pieces as many times as they wanted. The participants could take breaks freely during the experiment. The experiment took about 40 min.

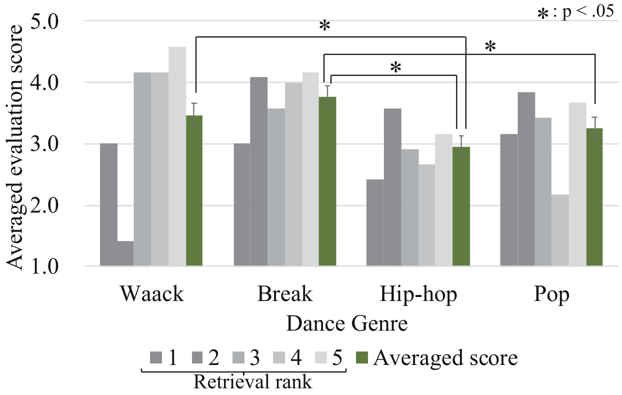

Fig. 6. The averaged Q1 scores for each dance genre.

Results and Discussion. Figure 6 shows the averaged Q1 scores for each dance genre. The vertical axis indicates the average of the Q1 scores from all of the participants, and the vertical bars indicate standard errors. The horizontal axis represents dance genres. The gray rectangles indicate the averaged evaluation scores for each retrieval rank. Each green rectangle indicates the average of all evaluation scores within the genre. We assessed the difference between the average Q1 scores with ANOVA. There was a significant difference ($F_{(3,236)} = 3.92, p < .05$). We also assessed the difference with Fisher's Least Significant Difference (LSD) test and found significant differences ($p < .05$) between waack and hip-hop, break and hip-hop, and break and pop.

Table 2 shows the Q1 evaluation scores and dance genres of each retrieval rank for each retrieval method. The hip-hop video had a comparatively bad performance as the query. The system returned musical pieces in the break genre. There are two reasons for this. Hip-hop is divided into more dance subgenres. The

Table 2. Top 5 retrieval results by dance genre. P in the table stands for pop, W for waack, and B for break.

Retrieval method	Dance genre	Retrieval rank				
		1	2	3	4	5
ADD (weighted)	Waack	W	P	W	W	W
	Break	B	B	B	B	B
	Hip-hop	B	B	B	B	B
	Pop	P	P	P	P	P

style of the hip-hop in our input video was middle hip-hop, but the hip-hop dance styles in the database were style hip-hop, girls hip-hop, jazz hip-hop, etc., dance styles slightly different from the query. Therefore, it was hard for the system to extract the musical pieces of hip-hop. The other reason is that middle hip-hop has some movements similar to those of break. One characteristic of break is when the dancer continues to dance while keeping their hands on the floor. However, before keeping their hands on the floor, breakdancers' movements are similar to middle hip-hop. Therefore, the system selected musical pieces for break. On the other hand, the system could extract motions similar to the query, which prevented the evaluation scores from markedly decreasing. For these two reasons, the system extracted musical pieces for break that were not the same dance genre of the query. In the future, we will divide the dance genres even further and add more dance genres into the database, which will improve evaluation scores.

The evaluation score for pop was worse than that of break and tended to be worse than that of waack. We interviewed the participants who gave a low score to pop to determine the reason for this. They said they scored it thus because the dance motions used as the query included a "vibration" technique in which the dancers move their bodies with a rapid trembling motion and a "wave" technique in which the dancers move their bodies like a wave. Those movements match specific sounds, and the participants decided that musical pieces that did not include those sounds were inappropriate for those movements. We can solve this problem by using interactive retrieval methods that let dancers adjust parameters according to their purposes. For example, if dancers want to search for specific musical pieces used in videos containing movements that closely resemble particular movements (like "waves"), the system will allow the dancers to search through a narrow range of musical pieces by adjusting parameters to match highly similar movements. In addition, if users want to search for musical pieces to practice to or to dance to in a club with many other dancers, the system will allow the users to search through a large range of various musical pieces by adjusting parameters to match the movements with a lower similarity. Users changing the parameters contextually could realize more efficient dance music retrieval.

5 Conclusion

We proposed Query-by-Dancing, which is a dance music retrieval system that enables a user to retrieve a musical piece using dance motions. We confirmed that the system's retrieval method is appropriate for dance music, that the system can find musical pieces that are easy to dance to, and that better music can be obtained by weighting the importance of dance motions when retrieving videos. Moreover, we conducted comparative experiments on 4 dance genres and confirmed that the system scored an average of 3 points or more evaluation points for 3 dance genres (waack, pop, and break), and our method can adapt to different dance genres. In the future, we plan to add a wider range of dance genres to the database of dance videos.

Acknowledgments. This work was supported in part by JST ACCEL Grant Number JPMJAC1602, Japan.

References

1. Cao, Z., Simon, T., Wei, S., Sheikh, Y.: Realtime multi-person 2D pose estimation using part affinity fields. In: The 2017 IEEE Conference on Computer Vision and Pattern Recognition (2017)
2. Casey, M.A., Veltkamp, R.C., Goto, M., Leman, M., Rhodes, C., Slaney, M.: Content-based music information retrieval: current directions and future challenges. Proc. IEEE **96**(4), 668–696 (2008)
3. Chen, J., Chen, A.: Query by rhythm: an approach for song retrieval in music databases. In: Proceedings of the 8th International Workshop on Research Issues in Data Engineering: Continuous-Media Databases and Applications, pp. 139–146 (1998)
4. Ghias, A., Logan, J., Chamberlin, D., Smith, B.C.: Query by humming - musical information retrieval in an audio database. In: Proceedings of ACM Multimedia 1995, pp. 231–236 (1995)
5. Jang, J.-S.R., Lee, H.-R., Yeh, C.-H.: Query by tapping: a new paradigm for content-based music retrieval from acoustic input. In: Shum, H.-Y., Liao, M., Chang, S.-F. (eds.) PCM 2001. LNCS, vol. 2195, pp. 590–597. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45453-5_76
6. Maezawa, A., Goto, M., Okuno, H.G.: Query-by-conducting: an interface to retrieve classical-music interpretations by real-time tempo input. In: The 11th International Society of Music Information Retrieval, pp. 477–482 (2010)
7. Müller, M.: Fundamentals of Music Processing - Audio, Analysis, Algorithms, Applications. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21945-5
8. Salvador, S., Chan, P.: Toward accurate dynamic time warping in linear time and space. J. Intell. Data Anal. **11**(5), 561–580 (2007)
9. Schedl, M., Gómez, E., Urbano, J.: Music information retrieval: recent developments and applications. Found. Trends Inf. Retr. **8**(2–3), 127–261 (2014)

10. Smiraglia, R.P.: Musical works as information retrieval entities: epistemological perspectives. In: The 2nd International Society of Music Information Retrieval, pp. 85–91 (2001)
11. Turnbull, D., Barrington, L., Torres, D., Lanckriet, G.: Semantic annotation and retrieval of music and sound effects. IEEE Trans. Audio, Speech, Lang. Process. **16**(2), 467–476 (2008)