

DanceDJ: ライブパフォーマンスを実現する実時間ダンス生成システム

岩本 尚也^{*†} 加藤 卓哉^{*} 柿塚 亮^{*} Hubert P.H. Shum[‡] 原 健太[§] 森島 繁生[¶]

概要. 本研究では、3D キャラクタの実時間ダンスパフォーマンスを実現するために、任意に流れる楽曲にマッチしたダンスを即興で演出するシステム、DanceDJを提案する。質の高いダンスを誰でも演出できるように、事前に用意したダンスデータベースを使用しながら、異なるダンス同士を滑らかに繋ぐことで一連のダンスを表現する。ダンスの操作インターフェースとして、ディスクジョッキー (DJ) ツールを採用することで、DJが楽曲をビートで扱うように、ダンスをビートで管理しながら異なるダンス間を繋いでいく体験を可能にする。なお本システムは、異なるダンス同士を自然に繋ぐための工夫として Transition Functionを導入し、その結果をビジュアルガイドダンスを通じてユーザに適切なダンスの切替タイミングを提示するように設計されている。最後に、鑑賞者及びユーザ視点の評価実験を実施し、システムの有効性を示す。

1 はじめに

マリオネットの紐に始まり、3DCGアニメーションのボーンに至るまで、キャラクターを実時間で操作するインターフェースは、長年用途に合わせて開発されてきた。こうしてキャラクターを音楽に合わせて実時間でダンスを踊らせるコンテンツは元来人形劇などで親しまれてきたが、現在ではヒューマノイドロボットやバーチャルアイドルのパフォーマンスなど幅広い発展が期待されている。しかしながら、従来のライブ向けの映像制作はコンピュータ上で予め用意されたダンスを対応する曲ごとにオフラインで作成する方法が主となっており、人形劇などで行われていたアドリブ表現ができず、観客と操作者とのインタラクティブ性が失われつつある。近年になって、R3[1]などの実時間でキャラクターを操作するシステムが試験的にステージに取り入れられている一方、多くのシステムはポーズや短いダンスをボタンに割り振っただけのインターフェースである。そのため、次のポーズやダンスの選択を支援する機能や、音楽のシンクロ性を考慮する機能など、ダンスを操作する上で必要な機能性に乏しく、操作性や生成されるダンスの完成度が低かった。

そこで我々は、ダンスの自動生成手法 [12][6]において広く用いられる尺度である、ダンス間の姿勢の類似性とダンスと音楽それぞれのビートを分析し提示することでダンス操作を効果的に支援できるという仮説を立てた。この仮説を検証するため、既存のダンスシーケンスを上記の支援に基づきインタラクティブにダンスを繋ぎ合わせることができると

システム”DanceDJ”を開発し、操作感や作成されたダンスについて評価する。ダンスと共通点が多い音楽において、DJが自ら楽器を演奏するスキルが無くとも楽曲をミックスすることで楽曲を実時間で編集できる点に着想を得て、本システムではDJツールを操作インターフェースとして採用した。ダンスをビートで管理しながら、異なるダンスシーケンスを自然に繋ぐことで、ダンスが得意でないユーザでも高品質なダンスを実時間で編集できることを検証する。

2 関連研究

キャラクターの姿勢を制御する様々なインターフェースが提案されている。3Dキャラクターのアニメーション生成において用いられる一般的な手法として、モーションキャプチャセンサやビデオ映像、またデプスセンサを用いたパフォーマンスキャプチャ法が挙げられる [3]。こうしたインターフェースでは、生成したい動きをユーザ自身で再現可能であることが前提となるため、ダンスのような高度な動作に対しては使用可能なユーザが制限されてしまう [8]。一方、Choiらの手法 [2]に代表されるスケッチベースのキャラクターの姿勢編集システムや、Glauserらの手法 [4]に代表される人形の入力デバイスインターフェースは直感的な姿勢編集が可能であるが、実時間でのダンス生成には適していない。実時間での姿勢制御の方法として、個々のモーションをボタンに割り当てて、ビデオゲームのようにキャラクターの動きを制御する手法 [14]もあるが、ダンスのバリエーションがボタンの個数に依存するため生成されるダンスのバリエーションに乏しい。このように、姿勢を編集するためのインターフェースの多くは実時間での操作性に乏しい上に、生成されるモーションの完成度が低いことがわかる。

本研究では、音楽を実時間で編集するインターフェースであるMIDIインターフェースを用いたインタラク

Copyright is held by the author(s).

* 早稲田大学

† 華為技術日本株式会社

‡ ノーザンブリア大学

§ 明治大学

¶ 早稲田大学理工学術院総合研究所

ションを提案する。MIDI コントローラを用いたシステムは、実時間での直感的な操作性に優れていることから、データの可視化やロボットのモーションコントロールを実現する研究 [11][9][5] など様々な研究において用いられている。ダンス生成においても、DJ ツールを用いたロボットのダンスモーション生成に関する研究 [13] では、実時間での楽曲や動きのテンポ制御に用いられている。しかしこの研究は、複数人で簡単にダンスが生成できることが目的とされており、ダンスと楽曲それぞれのビートや姿勢の類似度といったダンスの重要な要素を考慮しておらず、シンプルなダンスモーションしか生成できない。本研究ではより音楽での MIDI インタフェースの使用法を模倣することで、既存のシステムでは困難であった、実時間でのより複雑なダンスモーション編集を可能としている。また、ビートから解析されたダンス間の接続候補を提示することで、初めてシステムを使用する人にとっても、簡単に操作できる操作性を実現した。

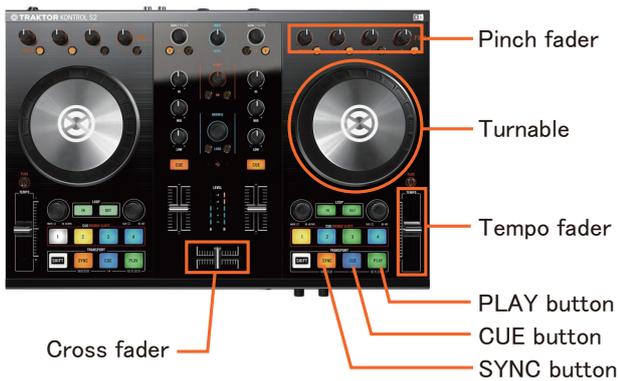


図 1. DJ インタフェースのボタンとスライダ

表 1. DJ インタフェースへ割り当てた機能一覧

インタフェース	ダンス制御パラメータ
テンポフェーダ	ダンスの再生スピード調整
クロスフェーダ	ダンス間のブレンド率調整
ピンチフェーダ	E_{pose} と E_{beat} への重み調整
ターンテーブル	前後数フレームの移動
プレイボタン	ダンスのスタート/ストップ
シンクボタン	楽曲とダンスのビート同期
キューボタン	ビジュアルガイドダンスの表示

3 DanceDJ のシステム概要

本システムで実現する CG キャラクタのライブパフォーマンスの流れについて述べる。まず、DJ が

現在プレイする楽曲のビート情報が *Ableton Link*³ を通じて DanceDJ のシステムに送られ、DanceDJ のユーザは受け取った楽曲のビートと一致するダンスを生成し、最後にそのダンスをプロジェクタで観客の前に投影する仕組みとなる。なお、ホログラム等を用いた 3 次元映像の出力も技術的に可能である。今回、DJ インタフェースとして用いた *TRAKTOR KONTROL S2*² の各ボタンやスライダに割り当てたダンスの制御パラメータは、既存の DJ システムに割り当てられている楽曲のパラメータを参考にしながら表 1 及び図 1 に示すような設計を行った。

以後、プロトタイプ段階で実装した DanceDJ の GUI の詳細及び DJ インタフェースの操作方法について図 2 と共に述べ、最後にプロトタイプ段階での課題について述べる。

3.1 ダンスの再生と補間

DanceDJ の GUI では、三体のキャラクタが横並びになっており、左右で異なるダンスモーションが割り当てられている。中央のキャラクタが踊るダンスは、左右のダンスをモーションブレンディングを適用して補間した結果であり、鑑賞者の前にも表示



図 2. (上) システムの GUI (左下) 観客に見える映像 (右下) DJ インタフェースを操作する様子

³ Ableton 社のビート同期システム

² TRAKTOR 社の 2 デック DJ コントローラ

される。その際のモーションのブレンド率は DJ インタフェースのクロスフェーダから取得した 0 から 255 の MIDI 信号が適用される。

3.2 テンポ調整

中央のキャラクタの下に表示されたバーは *Ableton Link* を通じて DJ から取得した楽曲のビート情報を示しており、赤色のバーはビートの 4 カウントの 1 カウント目を意味する。左右のキャラクタの下に表示されているバーは、左右のキャラクタに割り当てられたダンスの元の楽曲のビート情報となっている。なお、本研究で使用したダンスデータは、MikuMikuDance⁴上で様々なアマチュアクリエイタによって作成されたアニメーションデータが元になっているため、アーティストがダンスを作る上で参考にしたダンス動画の元の楽曲が存在する。ユーザが DJ の流している楽曲にシンクロするようにダンスの再生速度を調節する際には、テンポフェーダを用いて手で速度を調整することができる他、シンクボタンを押すことで自動的にビートを一致させる事ができるように設計されている。このビートの自動同期を実現するために、選択したダンスの元楽曲のビート情報を *Songle*⁵を用いて取得した後、単位時間あたりのビートの回数を計算することで BPM を求める。求めた BPM と DJ から受け取った楽曲の BPM の二つの情報から、楽曲の BPM と一致するダンスの再生速度を得ることが可能となる。

3.3 タイムライン上の移動

キャラクタの上部には、元の楽曲のタイトル、長さ、現在の時間、そして楽曲構造が表示されるため、ユーザはその情報から現在再生しているダンスが全体のどこに位置するかを把握できるようになっている。ターンテーブルを使えば手でキャラクタの再生位置を調整でき、次に繋ぎたいダンスの姿勢を探る際に頻りにターンテーブルが用いられる。本システムでダンスを繋ぐ際には、クロスフェーダを一方に寄せている間にもう一方のキャラクタに新たなダンスをロードし、ターンテーブルで繋ぎ先となるダンスの姿勢を探した後、クロスフェーダをスライドさせることで一連のダンスモーションを生成していく。

3.4 プロトタイプ段階での課題

上記に示した設計に基づいたプロトタイプを実装し、その使用感についてのユーザに簡単なヒアリングを行った結果、「どのタイミングでダンスを接続すべきかわからない」「左右のキャラクタ同士の姿勢を補間した結果が不自然になる場合がある」といった声が上がった。モーション生成に関する先行研究によって異なるモーション同士を自動で自然に

繋ぐことも技術的に可能であるが、我々の研究では、ユーザがダンスを繋ぐタイミングを自由に選べるインタラクティブ性を重視し、後述するダンスの姿勢やビートを考慮して適切な繋ぎのタイミングを提示する Transition Function, そしてその結果をユーザに提示するビジュアルガイドダンスの 2 つの機能を追加した。

4 Transition Function

本節では、異なるダンス間を自然に繋ぐための評価関数 Transition Function について述べる。本関数はユーザに対し、複数考えられる接続のタイミングの内、適切な繋ぎのタイミングをリファレンスとして提示できるように設計されているため、自然な繋ぎを実現するだけでなく、繋ぎのタイミングを選ぶ時間も短縮できる。本関数は接続のタイミングとして、式 (1) に示すような二つの項: (1) ダンスのポーズ間の類似度 $E_{pose}(i, j)$, (2) ダンスと楽曲のビートの類似度 $E_{beat}(i, j)$ を考慮する。

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

ここで、 i と j はそれぞれ異なるダンスの元楽曲のビートのインデックスを示す。また、 w_1 と w_2 は、評価関数 $E_{pose}(i, j)$ と $E_{beat}(i, j)$ に対する 0.0 から 1.0 の値を持つウェイトを意味する。これらのウェイトは、DJ インタフェースに備え付けられたピンチフェーダを用いて調整できる。以後、 $E_{pose}(i, j)$ と $E_{beat}(i, j)$ の具体的な計算手法について述べる。

4.1 姿勢間の類似度

異なるダンスを自然に繋ぐために、異なるダンスのペアデータに対して各フレーム間での姿勢の類似度を計算する。姿勢の類似度は、Kovar ら [7] と同様に、全ての関節に対する各関節位置との誤差の総和をフレームごとに計算し、計算結果となる行列の値を白黒の二次元テクスチャとして保存する。テクスチャの X 軸と Y 軸は各ダンスのフレーム番号を示し、白黒のピクセルは対応するフレームのペアでの姿勢の類似度の高低を示す。特に白いピクセルは姿勢の類似度が高いことを意味する。本類似度計算は事前計算として事前に用意した 17 個のダンスモーション全ての組み合わせに対して行われる。そして実時間処理として、現在与えられている二つのダンスのペアに対して事前計算されたテクスチャを読み込み、対応するフレーム位置でのピクセル値を呼び出すことで、類似度の値 $E_{pose}(i, j)$ を得る事ができる。

4.2 ダンスと楽曲のビートの類似度

ダンスを繋ぐタイミングとして、振り付けの途中はダンスを補間するタイミングとして適切ではないと考えられる。Okada らの研究 [10] では、一連のダンスアニメーションを振付ごとに分割するために、

⁴ 無償の 3DCG アニメーション制作ツール

⁵ 産業技術総合研究所による音楽解析及び可視化システム

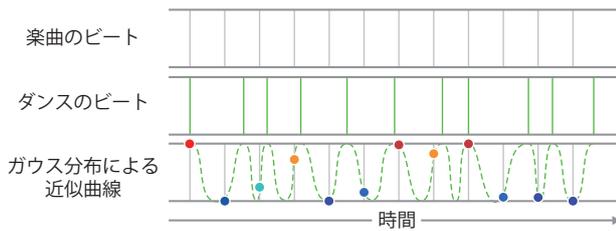


図 3. 楽曲とダンスのビート類似度計算イメージ図

楽曲のビート情報だけでなく、ダンスの情報を考慮する必要があると述べられている。そこで我々は、ダンスは身体で音楽のビートを表現する行為の一つであるといった仮定から、振り付けが切り替わるタイミングを考慮するために、ダンスのビートと楽曲のビートの類似度を評価する関数 $E_{beat}(i, j)$ を提案する。ダンスのビート計算には、Shiratoriら [12] が提案するモーションリズムの計算と同様に、各フレームごとに全ての関節の角速度の合計を用いた Weight Effort を計算し、あるフレーム幅ごとに値が極小となるフレームをダンスのビートとして定義する。ここで使用するフレーム幅は、各ダンスデータの元の楽曲から *Songle* の API を通じて取得した楽曲のビート幅を用いた。その計算結果の例をイラストとして図 3 に示す。この図のようにダンスから得られるビートは楽曲のビートのように必ずしも一定ではないため、我々はダンスと楽曲のビートの類似度を評価するために、楽曲のビートのタイミングでダンスのビートがどの程度一致しているかを考えることにした。この時、ダンスのビートのタイミングを 1 とする正規分布を考え、各ビートのタイミングでの正規分布を互いに足し合わせることで連続的な関数を求め、その連続したダンスのビート情報から楽曲のビートのタイミングでのダンスのビートの値を取得した。なお、計算時の正規分布のシグマの値は 0.1 とした。この処理はデータベース中の各ダンスデータに対して事前処理として行い、実時間では、評価する二つのダンスのビート i とビート j におけるビートの類似度値を足した値を $E_{beat}(i, j)$ とする。

5 ビジュアルガイダンス

本節では、異なるダンス間での適切な繋ぎのタイミングをユーザに提示するためのビジュアルガイダンスについて述べるために、図 4 に示すような左側に表示されるダンス A から右側のダンス B へ繋ぐタイミングを例として考える。ユーザがダンス B を制御する DJ ツールの右側のキューボタンを押すと、ダンス B がそのフレームで停止し、キャラクターの下に青から赤色へのグラデーションカラーのバーが表示される。各カラーバーの間隔はダンス B の元楽曲のビートの間隔を示しており、各バーの色は、4 章で



図 4. ビジュアルガイダンスを ON にした際の様子

述べた Transition Function の数値結果を色として表示している。なお、赤色が繋ぐタイミングとして適切であり、青色が適切でないことを意味している。図 4 の例では、Transition Function $E_{trans}(i, j)$ の i が常に更新されている A のビートのタイミングを、 j は静止したダンス B のビートのタイミングを意味しているため、ダンス A のビートが更新されるペースで右下のカラーバーが更新されていく。ユーザはこの様子から数ビート先の適切なタイミングを確認できるため、繋ぐタイミングを自身で選ぶことができる。繋ぐタイミングが来た際には、プレイボタンを押して、ダンス B を再生すると同時にクロスフェーダを移動させることで滑らかな繋ぎを実時間で実現できる設計となっている。なお、右下のカラーバーでは、数ビート先の局所的な情報しか見ることができないため、大局的な情報を見れるようにキャラクター A の上に全てのビートのタイミングでの E_{trans} の値を表示した。ユーザがキューボタンを押した際に静止したキャラクター B のポーズはターンテーブルで前後数フレームを確認でき、移動したフレームでの E_{trans} の値を反映した局所的及び大局的なビジュアルガイダンスが同時に更新される。

6 評価実験

本論文で提案したシステムの有効性を検証するために、鑑賞者から見た生成結果と、ユーザとしてシステムを使用した際の使用感について、評価実験を行った。その際、鑑賞者視点における評価では 17 人 (男性 15 人, 女性 2 人), ユーザ視点における評価では 12 人 (男性 11 人, 女性 1 人) の被験者を集めた。被験者の中でダンス経験のある人は、鑑賞者視点における評価では 4 人, ユーザ視点における評価では 3 人含まれていた。

表 2. 鑑賞者への質問 (7 段階評価, 7:強くそう思う; 1: 全く思わない)

番号	質問
Q1.	生成結果に満足したか
Q2.	ダンス間の遷移は自然だったか
Q3.	生成結果が音楽とマッチしていたか

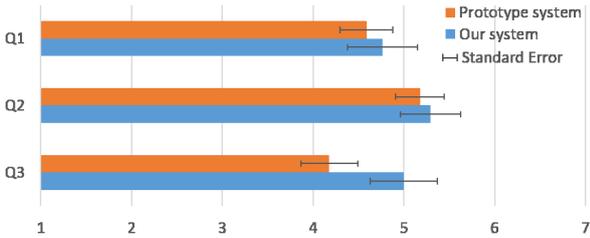


図 5. 鑑賞者視点でのプロトタイプとの比較結果

6.1 鑑賞者視点における評価

鑑賞者視点における評価では, DanceDJ 経験のあるユーザがプロトタイプと本論文で提案したシステムそれぞれで楽曲に合わせて生成した 2 種類のダンスを各被験者に数分間鑑賞してもらった後, 表 2 のアンケートに回答してもらった. その結果を図 5 に示す. 各システムの評価値を Wilcoxon の符号付き順位検定により比較したところ, 全ての質問項目について有意差は認められなかったものの, 標本内での平均値で比較すれば提案システムがプロトタイプシステムよりも高い評価を得ている. このことから, 提案システムは高品質のダンスアニメーションを生成するために有効に機能し得るということが分かった.

6.2 ユーザー視点における評価

ユーザー視点における評価では, まず被験者にプロトタイプと提案システムの使用方法を 5 分間で説明した後, DJ の再生する楽曲に合わせてユーザとして 10 分間プレイしてもらい, 最後に, 表 3 のアンケートに回答してもらった. Q1 から Q3 までの結果を図 6 に, Q4 と Q5 の結果を図 7 に示す. 各システムでの Q1 から Q3 までの質問項目における回答を, 危険率を 1% として Wilcoxon の符号付き順位検定により比較した結果, 提案システムはプロトタイプに比べて評価値が有意に高かった. また Q4 の回答結果より, 提案システムでは各ボタンとモーションの対応関係として, 被験者全員が「5. 概ね適切だと感じた」以上の評価をしていることがわかる. さらに Q5 の回答結果から, 提案システムでは 90% 以上の被験者が 10 分以内で満足できるダンスアニメーションが作れるようになっていることがわ

表 3. ユーザへの質問 Q1-Q4(7 段階評価, 7:強くそう思う; 1: 全く思わない), Q5(選択式)

番号	質問
Q1.	ダンス間を自然に繋ぐことができたか
Q2.	ダンスと音楽を同期できたか
Q3.	ダンスの生成結果に満足したか
Q4.	各ボタンへの機能の割当ては適切か
Q5.	満足なダンスを作るまでの時間

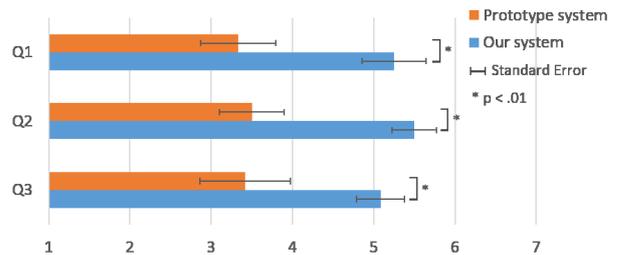


図 6. ユーザ視点でのプロトタイプとの比較結果 (Q1-3)



図 7. ユーザ視点でのプロトタイプとの比較結果 (Q4-5)

かる. 以上のことから, DJ ツールを用いた提案システムで直観的な実時間ダンス生成が可能であることが示された. また, ユーザに対し, 接続の際にどのような情報を参考にしたかヒアリングしたところ, プロトタイプでは上位からキャラクターの姿勢, 楽曲情報, ビート情報の順になったが, 提案システムでは, ビート情報, 接続性, 楽曲情報という順になった. 本結果から, プロトタイプ段階では接続に際し, ビート情報を重視していないのに対し, 提案システムでは我々の提案したビートでダンスをコントロールするという仕組みにユーザも適応できていると言える.

6.3 その他のフィードバック

我々は, 実際のライブシーンにおける聴衆のフィードバックを得る為にビジュアルジョッキー (VJ) イベントに参加し, そのライブ会場において約 30 人程度の観客の前で 20 分間の実演を行った. イベント中にプロジェクトの映像に本システムのキャラク

タが表示された瞬間、観客から驚きの歓声が沸き、楽曲とダンスのビートが一致した際には、観客から「(タイミングが) 合った!」といった興奮混じりの声を聞くことができた。また VJ が入れ替わる際に、DanceDJ ユーザと次を担当する VJ ユーザ間で互いの映像を重ね合わせることでできた CG キャラクタがステージの上で踊っているような演出を偶発的に見ることができた。今後、DJ と VJ と DanceDJ 間の連携による幅広い演出が期待される一幕となった。

7 終わりに

我々は、3D キャラクタの実時間ダンスパフォーマンスを実現するために、DJ が任意に流す楽曲のビートに同期したダンスを即興で演出するシステム”DanceDJ”を提案した。本システムを実現する上で課題となる異なるダンス間の滑らかな補間を Transition Function 及びビジュアルガイダンスによって実現し、評価実験を通じて有効性を確認した。

今後は、より鑑賞者の評価を向上させるために、異なるダンス同士を自然に繋ぐだけではなく、繋ぐタイミングでキャラクタを変更するなどの仕組みを考える他、キャラクタの演出として、カメラワークの考慮や画面を鮮やかに彩るエフェクト等を追加する予定である。エフェクトに関しては、VJ との共演を実現することでダンスと映像による相乗効果が生まれるのではないかと考えている。また、DJ が楽曲の低音や高音の強弱を変化させるように、楽曲の盛り上がりに応じてユーザがモーションフィルタ等を加えることで、再生するダンスをより感情的にするといった演出も考えている。

8 謝辞

本研究は、JST ACCEL No.JPMJAC1602 の援助により実施された。また本論文の著者である岩本尚也及び加藤卓哉は筆頭著者として同等に貢献した。

参考文献

- [1] R3-リアルタイム 3DCG コントロールシステム。 <http://r3system.net/>.
- [2] B. Choi, R. B. i Ribera, J. P. Lewis, Y. Seol, S. Hong, H. Eom, S. Jung, and J. Noh. SketchiMo: Sketch-based Motion Editing for Articulated Characters. *ACM Trans. Graph.*, 35(4):146:1–146:12, jul 2016.
- [3] A. Fender, J. Müller, and D. Lindlbauer. Creature Teacher: A Performance-Based Animation System for Creating Cyclic Movements. In *Proceedings of the 3rd ACM Symposium on Spatial User Interaction, SUI '15*, pp. 113–122, New York, NY, USA, 2015. ACM.
- [4] O. Glauser, W.-C. Ma, D. Panozzo, A. Jacobson, O. Hilliges, and O. Sorkine-Hornung. Rig Animation with a Tangible and Modular Input Device. *ACM Trans. Graph.*, 35(4):144:1–144:11, jul 2016.
- [5] P. Groth and D. A. Shamma. Spinning Data: Remixing Live Data Like a Music Dj. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems, CHI EA '13*, pp. 3063–3066, New York, NY, USA, 2013. ACM.
- [6] R. Kakitsuka, K. Tsukuda, S. Fukayama, N. Iwamoto, M. Goto, and S. Morishima. Authoring System for Choreography Using Dance Motion Retrieval and Synthesis. In *The 30th International Conference on Computer Animation and Social Agents (CASA 2017)*, pp. 122–131, 2017.
- [7] L. Kovar, M. Gleicher, and F. Pighin. Motion Graphs. *ACM Trans. Graph.*, 21(3):473–482, jul 2002.
- [8] Z. Liu, J. Huang, S. Bu, J. Han, X. Tang, and X. Li. 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.
- [9] A. Norman and X. Amatriain. Data jockey, a Tool for Meta-Data Enhanced Digital DJing and Active listening. In *ICMC*. Michigan Publishing, 2007.
- [10] N. Okada, N. Iwamoto, T. Fukusato, and S. Morishima. Dance Motion Segmentation Method Based on Choreographic Primitives. In *Proceedings of the 10th International Conference on Computer Graphics Theory and Applications, GRAPP 2015*, pp. 332–339, Portugal, 2015. SCITEPRESS - Science and Technology Publications, Lda.
- [11] M. M. Ragnhild, M. Mckelvin, R. Nest, L. Valdez, K. ping Yee, M. Back, and S. Harrison. 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.
- [12] T. Shiratori, A. Nakazawa, and K. Ikeuchi. Dancing-to-Music Character Animation. *Comput. Graph. Forum*, 25(3):449–458, 2006.
- [13] T. Shirokura, D. Sakamoto, Y. Sugiura, T. Ono, M. Inami, and T. Igarashi. 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, ACE '10*, pp. 53–56, New York, NY, USA, 2010. ACM.
- [14] Y. Yazaki, A. Soga, B. Umino, and M. Hirayama. Automatic Composition by Body-Part Motion Synthesis for Supporting Dance Creation. In *2015 International Conference on Cyberworlds (CW)*, pp. 200–203, Oct 2015.