ダンスフォーメーション作成のための候補提示型インタフェース

遠藤 輝貴 * 盧 承鐸 * 中島 一崇 * 楊 溪 * 五十嵐 健夫 *

概要. 集団でのダンス作品において複数人で構成されるフォーメーションや移動は重要な要素であり，振付師は紙に図を書いて様々な案を検討したり説明したりするが，静的な図ではその理解が難しい．解決策としてフォーメーションや移動を可視化するアニメーションを生成することが挙げられるものの，各ダンサーの動きを一つ一つ指定するには時間と手間がかかってしまう．そこで本研究では，ダンスのフォーメーションや移動経路を効率的に設計するための「候補提示型」ユーザインタフェースを提案する．具体的には，ユーザが少数のダンサーの移動経路を入力すると，本システムは残りのダンサーの移動経路をルールベースで複数予測する．これらの候補は計算された優先度の高いものから順にサムネイル形式で提示され，ユーザはその中から一つを選択することで残りのダンサーの経路をまとめて生成する．本研究では 4 人の参加者に対してユーザテストを実施し，提案システムがフォーメーション移動の作成に役立つことを確認した．

1 はじめに

複数人でのダンスパフォーマンスにおいて，フォーメーションやその移動は観客を惹きつけるのに重要である．ダンス作品を作る振付師はフォーメーションや移動を紙に書いて考えることが多いが，紙に書かれた静的な図だけでは各ダンサーがいつ，どのように動くかの理解が難しい．その結果，振付師は各ダンサーの動きを把握しきれず，ダンサー同士の衝突などを含んだ移動を作ってしまう可能性がある．また，紙の図だけでは作った移動を各ダンサーに説明することも難しく，全員が移動を理解するために何度も練習をする必要がある．

この問題の解決策として，フォーメーションや移動のアニメーションを作るということが挙げられる．移動をアニメーションで可視化することにより，振付師やダンサーが移動の仕方を理解することが容易になり，ダンサー同士の衝突の有無も視覚的に確認できるようになる．このような移動アニメーションの作成をするためのアプリケーション [1, 18, 19] や研究 [7, 17] はいくつか存在するが，それらは基本的にユーザが移動を一つ一つ作っていく必要がある．ChoreoGraphics [16] ではダンサーの移動経路を一括で生成できる機能があるが，生成できる移動は単純なものに限られており，さらに移動のルールをユーザが明示的に指定する必要があった．

そこで我々はダンスのフォーメーションや移動の作成を支援するための，候補提示型インタフェースを備えたシステムを提案する．本システムでは，ユーザはまず移動前後のフォーメーションを指定し，少数のダンサーの移動経路を作成する．すると本システムはヒューリスティックなルールに基づき，移動前後の位置の対応関係とその間の移動経路を自動で予測する．ユーザの入力に曖昧性があることを考慮して，本システムは移動の候補を複数予測し，それらをサムネイル形式のリストにしてユーザに提示する．ユーザはその中から一つを選択することで，全体のフォーメーション移動をまとめて作成できる．

本システムに備えられた候補提示型のインタフェースには以下の利点がある．まず，ユーザはシステムが提示した移動の候補を選択するだけで簡単に全体の移動を作成できる．これは移動するダンサーの数が多くなった時に特に有効である．次に，本システムはユーザが描いた経路から残りのダンサーの移動経路を自動で予測する．そのため，候補生成のためにユーザが追加で操作を行う必要はなく，簡単に利用できる．また，本システムは複数の移動候補を生成して提示するため，ユーザである振付師の意図していなかった移動も提示される可能性がある．このことは振付師が今まで思いつかなかった新しい移動のアイデアを得るのに役立つ．最後に，本システムは単に移動の候補をユーザに提示するだけであり，その候補を採用することをユーザに強制しない．そのためシステムがユーザの自由な制作を妨げることはなく，ユーザは必要な時だけ提示された候補を使うことができる．

提案システムの評価のため，4 人の振り付け経験者に対してユーザテストを行い，本システムを用いてフォーメーションや移動を作成してもらった．実験で作られた移動経路のうち約 63%は本システムによって提示されたものであり，さらに参加者から移動のアニメーションを簡単に作成できて便利だという回答を得られたことから，本システムの有効性，有用性を確認した．

なお，本稿は国際会議で採択された内容 [2] をより詳細に記述したものである．

Copyright is held by the author(s).

* 東京大学

2 関連研究

2.1 ダンスフォーメーション作成

グループダンスのフォーメーションとその移動を作成するためのアプリケーション [1, 18, 19] や研究 [7, 17] はいくつか存在するが，これらはユーザが各ダンサーの位置や移動経路を一つ一つ指定する必要があり，特にダンサーの人数が増えると作成に手間がかかる．ChoreoGraphics [16] ではユーザが移動前後のフォーメーションを作成してその間の移動ルールを指定すると，各ダンサーの移動経路をまとめて作成できる機能がある．ただしどの移動ルールを使用するかはユーザが明示的に指定する必要があり，ルールの種類も限られていて直線的な移動しか作成できない．一方で Lee と Lee [8] はダンスのフォーメーションと移動を全て自動で生成するシステムを提案・開発しているが，このシステムではユーザが移動を直接制御することはできない．我々のシステムではユーザが自由に移動経路を作成でき，候補の提示によりその作成を支援することを目的としている．ユーザは少数のダンサーの移動経路を作成するだけでシステムが残りのダンサーの経路を自動で予測して作成するため，移動のルールを指定しなくても簡単に移動経路を作成できる．また本システムは移動の候補を一度に複数生成するため，ユーザの意図していない移動候補も提示される可能性があり，新しい移動のアイデアを得ることにも役立つ．

また，集団の動きを制御するための研究はクラウドシミュレーションの分野でも行われている [3, 4, 6, 9, 15, 20]．これらの研究は大人数の集団の全体的な動きを制御することに主眼が置かれているが，我々が扱うダンスのフォーメーション作成では比較的少ない人数を対象とし，各ダンサーの移動を個別に，より精細に制御することを目的としている．

2.2 候補提示型インタフェース

候補提示型インタフェースとはユーザから与えられたヒントをもとにユーザの望む操作を予測して複数の候補を生成しユーザに提示するインタフェースである．これは Igarashi と Hughes [5] によって提案され，2D [13] や 3D [5] の図形描画，グラフィックデザイン [14]，家具の配置 [10] などに応用されている．これらのシステムはユーザに複数の候補をサムネイル形式で提示するため，ユーザはその中から望みの候補や自分では考えつかないような候補を簡単に見つけることができるという特長があり，本システムでもこの候補提示型の手法を採用している．

3 提案システムのインタフェース

3.1 フォーメーションの作成

本システムのインタフェースを図 1 に示す．本システムでは PowerPoint [11] のスライドのようにそれぞれのフォーメーションを別々のシーンに分けており，ユーザは各シーンに対してダンサーの位置を配置してフォーメーションを作成する．それぞれのシーンはダンスのステージを上から見た図となっており，シーン上にある実線の丸は現在のシーンでのダンサーの位置を，点線の丸は 1 つ前のシーンでの位置を表す．なお，現在のシステムではダンサーの向きについては考慮していない．ユーザは左のシーンリスト（図 1a）から編集するシーンを選び，上部のボタン（図 1b）で Select，Drawing，Path の 3 種類のモードを切り替えて，真ん中のメインキャンバス（図 1c）でシーンを編集していく．

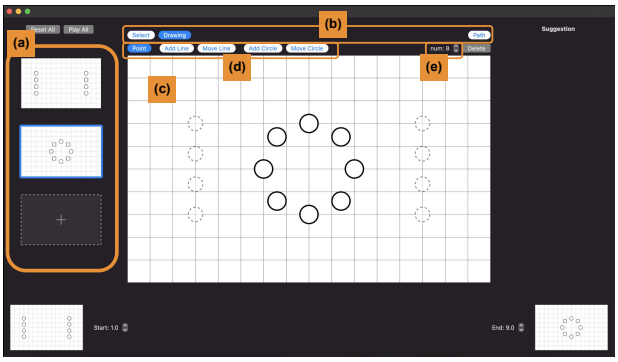


図 1. Drawing モードの UI．(a) シーンリスト．(b) モード切り替えボタン．(c) メインキャンバス．(d) ツールボタン．(e) ダンサー数変更ステッパー．

まず，ユーザは Drawing モードでダンサーを配置していく．ここでは，前後のシーンでのダンサーの位置の間に関係はなく，ユーザは位置の対応関係を意識する必要はない．このような実装としたのは，振付師が作品を作る際に移動前後のフォーメーションは決まっているがその間の対応関係や移動を決めていないという場合があり，そのような場合でもシステムを利用できるようにするためである．Drawing モードには図 1d に示すツールボタンがあり，Point ツールではクリックした位置にダンサーを一つ一つ配置できる．Add Line ツールではドラッグによって直線を引き，新たなダンサーの位置をその直線上に等間隔に配置することができる．描いた直線は Move Line ツールで位置や長さを変更でき，また図 1e に示すステッパーで直線上に配置するダンサーの数の変更も可能である．同様に，Add Circle，Move Circle ツールを使うとドラッグによって楕円を描き，その上に等間隔に新たなダンサーの位置を配置できる．また，Select モードでは既に配置したダンサーの位置をドラッグにより個別に動かして微調整ができる．

3.2 移動経路の作成

ダンサーの位置を配置して移動前後のフォーメーションを作成したら，ユーザは Path モードに切り

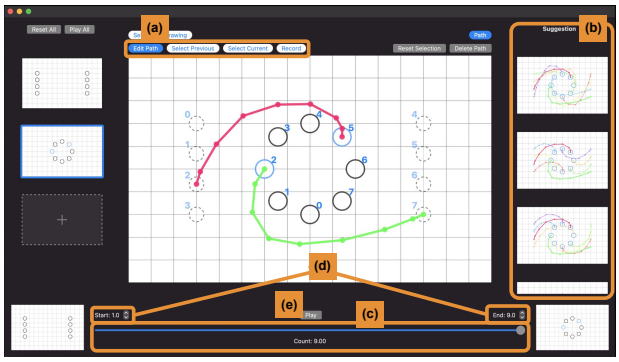


図 2. Path モードの UI．(a) ツールボタン．(b) 候補リスト．(c) シークバー．(d) 移動カウント指定用ステッパー．(e) アニメーション再生ボタン．

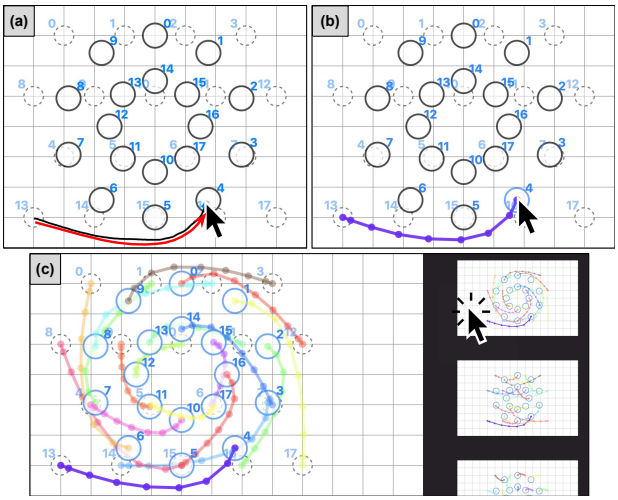


図 3. 本システムにおける移動経路作成の様子．(a) ユーザはドラッグでダンサーの移動経路を描く．(b) システムはドラッグの軌跡から経路を作成する．(c) システムによって提示された候補を選択するとその候補の移動がメインキャンバスに反映される．

替えて間の移動を作成する．このモードでは点線の丸で示される 1 つ前の位置から実線の丸で示される現在の位置への各ダンサーの移動を編集する．Path モードの UI は図 2 のようになっており，上部にツールボタン（図 2a），右側に候補リスト（図 2b），下部にシークバー（図 2c）が表示される．

ユーザは以下の手順で移動経路を作成する．まず，シークバーの端にあるステッパー（図 2d）で移動の開始と終了のカウントを指定する．カウントとはダンスにおける音楽の拍を意味し，現在のシステムでは 1 カウントが 0.5 秒に対応している．ダンサーは 8 カウントを 1 つの単位として扱うことが多いため，デフォルトでは移動にかかるカウント数を 8 としている．ユーザは次にダンサーの移動経路を描く．図 3a のようにユーザが 1 つ前の位置の中の 1 つから現在の位置の中の 1 つまでドラッグすると，描いた軌跡をシステムが記録して図 3b のように経路を作成する．経路は 1 カウント毎に頂点を持つ折れ線で表され，識別のために経路毎に別の色がついている．また，折れ線の頂点は端の頂点に近ければその頂点に自動で重なるようになっており，これによって移動開始や終了のタイミングをずらすことができる．経路の作成後に図 2e の Play ボタンを押すと，移動のアニメーションが再生される．このアニメーションでは各ダンサーは 1 カウント毎に今いる頂点から次の頂点へと折れ線上を移動する．また，経路上の頂点の位置はドラッグによる微調整が可能である．

以上のようにしてユーザは移動経路を作成することができるが，この方法でダンサー全員の経路を作成するのは時間と手間がかかる．その代わりに，本システムではコンピュータに経路を予測させることで複数ダンサーの経路をまとめて作ることを可能にしている．これを行うためには，ユーザはまず移動前後で同じ数だけのダンサーの位置を選択する．システムはここで選択された位置に対して経路の予測を行う．図 2a に示す Select Previous，Select Current ツールを使うと，それぞれ移動前，後のダンサーの位置の選択と解除をクリックで切り替えられる．選択された位置には 0 から始まる連続した番号がつけられるが，この番号が移動前後の位置の対応関係を表している訳ではなく，対応関係は番号順と関係なくシステムが自動で計算する．候補生成時に番号の順に位置を対応させる特別な場合もあるが，それについては第 4.2 節で述べる．なお，初期状態では配置された全てのダンサーの位置が選択されており，全員分の移動をまとめて作る場合は新たに位置を選択する必要はない．

その後，選択された位置の中の 1 つまたは複数に対してユーザが移動経路を描くと，システムは選択された残りの位置に対して移動経路を予測する．本システムの経路予測は 2 段階で行われ，まず最初に移動前後の位置の対応関係を計算し，次にダンサーが移動前の位置から対応する移動後の位置へと動く軌道を計算する．システムは移動の候補を複数生成し，図 3c のようにサムネイル形式のリストで提示する．これらの候補はアニメーションで動かすこともできるため，ユーザは各候補の移動の仕方を視覚的に確認できる．最後にユーザはこれらの候補から 1 つを選択することでその候補をメインキャンバスに反映し，移動の作成を完了する（図 3c）．ここで，システムによって予測された経路はユーザが描いた経路よりも色が薄くなっており，この経路が未確定であることを表している．ユーザがメインキャンバス上で経路を編集すると，システムは未確定の経路に対して予測を更新し，新たな候補を生成する．例えば図 4a ではオレンジ色の経路がユーザの描いた

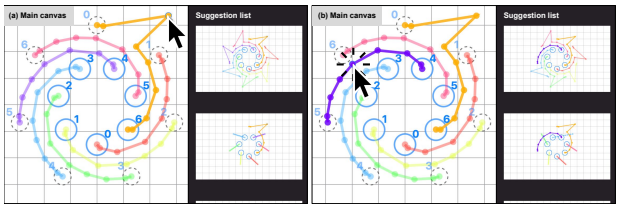


図 4．移動経路修正の様子．(a) ユーザがオレンジの経路を編集すると右側のリスト中で残りの未確定の経路に対する予測が更新される．(b) ユーザが紫の経路を編集すると経路が確定し，右側のリストで予測が更新されなくなる．

経路，残りの経路がシステムによって予測された経路であり，ユーザがオレンジの経路を編集するとそれに合わせて右側の提示候補も更新されている．図 4b のようにユーザが未確定である紫色の経路を編集するとその経路は確定状態となり，予測の中で紫の経路は変更されなくなる．

4　候補生成アルゴリズム

4.1　アルゴリズムの概要

本システムはユーザが描いた経路を入力として受け取り，残りのダンサーの移動経路を予測してユーザに提示する．システムが経路を予測する際はまず移動前後のダンサーの位置の対応関係を計算し，次いで移動前の位置から対応する移動後の位置へと移動する軌道を計算する．経路の予測は互いに独立な 5 つのヒューリスティックなルールを用いて行われ，各ルールによって 1 つまたは複数の候補が生成される．最後にシステムは生成した候補を整理してサムネイル形式のリストとしてユーザに提示する．

4.2　ヒューリスティックルール

本システムでは R1 から R5 の 5 つのヒューリスティックなルールを用いて移動経路を予測している．各ルールはそれぞれ異なる移動の仕方を想定しており，それぞれの想定に従ってダンサーの移動前および移動後の位置を変換し，変換後の位置の間の距離の 2 乗和を最小にするマッチングを求める．その後，マッチング結果に基づいて軌道を求める．以降では経路を予測するダンサーの数を N，番号 i の移動前の位置を p_i，移動後の位置を q_i とし，それぞれの変換後の位置を $f(p_i)$，$g(q_i)$ とする．さらに $\sigma : \{0, \ldots, N-1\} \to \{0, \ldots, N-1\}$ を $p_{\sigma(i)}$ から q_i への移動を表すマッチングとすると，各ルールが計算する最小マッチング m はコスト関数を $d(p, q) = |p - q|^2$ として

$$m = \arg\min_{\sigma} \sum_{i=0}^{N-1} d\big(f(p_{\sigma(i)}), g(q_i)\big) \qquad (1)$$

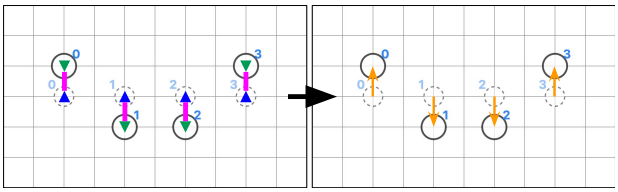


図 5．R1 による候補生成例．青い上三角が $f(p_i)$，緑の下三角が $g(q_i)$ を表し，紫の線が最小マッチング m を，オレンジの矢印が作成された経路を表す．

と表せる．コスト関数に距離の 2 乗を使うと単純な距離を使うよりもマッチング中の最大移動距離を小さくでき，1 人のダンサーのみが長い距離を動くような移動を防ぐことができる．なお，実装では最小コストマッチングの計算はハンガリアン法 [12] を用いて効率的に求めており，各候補は対話的な操作に差し支えない速度で生成できる．

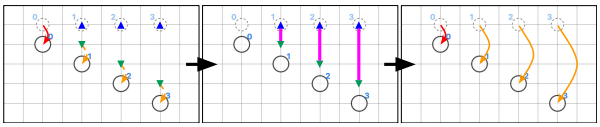
R1（図 5）は 5 つの中で最も単純なルールであり，各ダンサーの移動コストの総和が最小になるように $f(p_i) = p_i, g(q_i) = q_i$ としてマッチングを求める．次にシステムは求めたマッチング m を用いて，各 $i = 0, \ldots, N-1$ に対して $p_{m(i)}$ から q_i への経路を直線で作成する．R1 だけはユーザの描いた経路を使用していないため，ユーザが経路を 1 つも描いていない場合でも候補を生成できる．

R2 から R4（図 6）はユーザが描いた経路を使用して候補を生成する．もしユーザが複数の経路を描いた場合，システムはそれらの経路を 1 つずつ選び，各経路に対して候補を生成する．このとき $f(p_i) = p_i$ であり，$g(q_i)$ についてはユーザが描いた経路を残りの移動後の位置に対してコピーし，図 6 のオレンジ色の点線の矢印のように経路の終点が移動後の位置に重なるようにした時のコピーした経路の始点となる．システムはマッチングを求めた後，最初にコピーした経路を回転・拡大・移動することで，経路の始点と終点が移動前と移動後の点に一致するように変換する．以上のようにして，R2 から R4 はユーザが描いた経路の 1 つ 1 つに対して 1 つの候補を生成する．

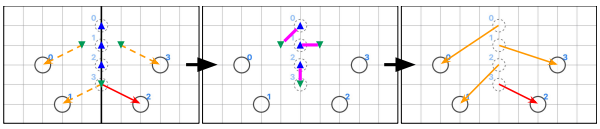
R2（図 6a）の場合はユーザが描いた経路の始点と終点をそれぞれ p_u，q_u として

$$g(q_i) = q_i - (q_u - p_u) \qquad (2)$$

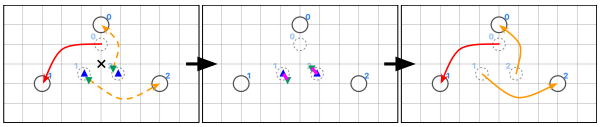
となる．これは直感的にはダンサー全員が同じ方向に移動することを想定しており，$g(q_i)$ はユーザが描いた経路をそのまま平行移動して残りの移動後の位置にコピーした時の始点に対応する．R3（図 6b）はダンサーの位置の中心を通り水平または垂直な軸に対して対称的な移動をすることを想定しており，ユーザが描いた経路をそのまま平行移動するか，軸を跨ぐ場合は反転してそれぞれの移動後の位置にコピーする．ダンサーの位置の中心 c は全ての p_i と



(a) R2．同一方向への移動を想定する．



(b) R3．黒い太線の軸を対称とした移動を想定する．



(c) R4．黒い × 印を中心とした回転移動を想定する．

図 6．R2 から R4 による候補生成例．赤い矢印がユーザの描いた経路を，左側のオレンジの点線の矢印がその経路のコピーを表す．その他は図 5 と同様．

q_i の平均を取ることで得られ，この点を通る軸を ℓ とし，ℓ に関して p_u，q_u と対称な点をそれぞれ p'_u，q'_u とすると，$g(q_i)$ は q_i と q_u が ℓ に関して同じ側のとき (3)，反対側のとき (4) となる．

$$g(q_i) = \begin{cases} q_i - (q_u - p_u) & (3) \\ q_i - (q'_u - p'_u) & (4) \end{cases}$$

R4（図 6c）はダンサーが中心 c の周りを回転するような移動を想定しており，ユーザが描いた経路を c の周りに回転，拡大して移動後の各位置にコピーする．このとき $g(q_i)$ は

$$g(q_i) = kR(p_u - c) + c \tag{5}$$

と表される．拡大係数 k と回転行列 R は以下の式から求められる．

$$k = \frac{|q_i - c|}{|q_u - c|}, \quad q_i - c = kR(q_u - c). \tag{6}$$

以上の 4 つのルールはダンサーの位置に振られた番号の順と関係なくマッチングを求めるので，例えば図 6c のように 2 番目の移動前位置から 0 番目の移動後位置へ移動することも可能である．

一方，R5（図 7）は各位置に振られた番号の順に一列になって動くような移動を想定するので，移動前後で位置の番号は同じになる．この場合，p_0 から q_0 へと動くダンサーが列の先頭になるので，ユーザはこの先頭のダンサーの移動経路を描く．するとシステムはこの経路の後ろに，番号順に並んだ p_i をつなげることで折れ線 $\mathcal{C}$ を作り，$\mathcal{C}$ を残りのダンサーの移動経路とする．ただし，経路の終点だけは $\mathcal{C}$ 上でなく各 q_i に重なるようにする．このように，R5

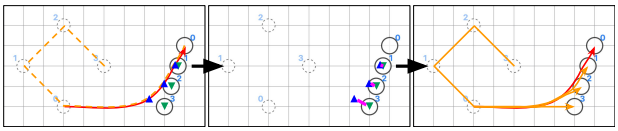


図 7．R5 による候補生成例．オレンジの点線が $\mathcal{C}$ を表す．その他は図 5，6 と同様．

はユーザが p_0 から q_0 への経路を描いたときのみ，候補を 1 つ生成する．R5 の場合はダンサーの位置のマッチングは番号によって既に決まっているが，それでもシステムは 4.4 節で述べる候補整理のためにマッチングコストに相当するものを求める必要がある．この場合は $g(q_i) = q_i$ とし，$f(p_i)$ は折れ線 $\mathcal{C}$ 上にあって以下の式を満たすものとして求める．

$$f(p_0) = q_0, \tag{7}$$

$$|f(p_i) - f(p_{i-1})| = |q_i - q_{i-1}|. \tag{8}$$

マッチングコストは $d\big(f(p_i), g(q_i)\big)$ の総和とする．

4.3 移動タイミングの予測

システムはこれらの 5 つのルールで生成された経路に対し，各ダンサーの移動開始や終了のタイミングも同時に予測する．ユーザは移動経路描画後，タイムカウントを表す経路上の頂点の場所を変更することで，移動の開始や終了のタイミングを設定することができる．システムは移動カウントが指定された経路をもとに，残りの経路の移動カウントを予測する．移動カウントの指定された経路が 1 本の場合には，そのカウントがそのまますべての経路に適用される．カウントの指定された経路が複数ある場合，移動後の位置の番号 i から移動カウント c_i を計算する一次関数 $c_i = ai + b$ を，指定されたカウントを使った最小二乗線形回帰によって求め，番号 i の位置の移動カウントを $ai + b$ に最も近い整数とする．例えば図 8a ではユーザが 0 番と 1 番のダンサーの経路を描いており，移動開始はどちらも 1 カウント目で，移動終了はそれぞれ 2，3 カウント目としている．システムはこれをもとに残りの i 番目のダンサーに対して移動開始が 1 カウント目，移動終了が $(i+2)$ カウント目であると予測し，経路を生成する．

4.4 候補の整理

あまりにも多くの候補を提示されるとユーザが困惑してしまうため，システムは最後に各ルールによって生成された候補を整理する．まず候補の中で同じ移動の仕方を生成しているものを 1 つにまとめ，その重複度が高い順に候補を並び替える．重複度が同じ候補については，計算したマッチングコストが小さい方をより高い順位とする．その後順位が高い 5 つの候補をリストにし，ユーザに提示する．

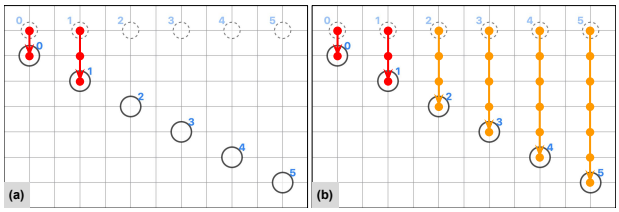


図 8. 移動タイミングの予測例．矢印の色の意味は図 6 と同様であり，1 カウント毎に通過する点を丸い頂点で示している．(a) ユーザの描いた経路．(b) システムが (a) をもとに予測した経路．

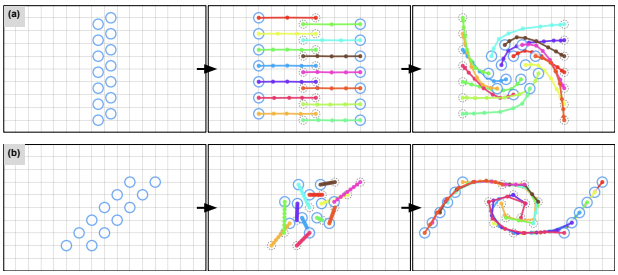


図 9. ユーザテストで用いた課題作品．(a) 課題 A．(b) 課題 B．

5 ユーザテスト

5.1 実験手順

本システムの有効性と有用性を調べるため，P1 から P4 の 4 人の参加者に対してユーザテストを行った．参加者は全員が 10 人から 30 人程度で踊るダンス作品の制作経験を持ち，実験前のアンケートでは P2 と P4 は主に紙を，P1 と P3 は紙と PowerPoint [11] などのアプリケーションを利用してフォーメーションや移動を考えていると答えた．本システムは Swift で実装したアプリケーションであり，参加者に事前にアプリケーションを送って各々のパソコンから起動してもらい，使用している様子をビデオ会議ツールで録画することによって，オンラインで実験を行った．所要時間は 1 人当たり約 1.5 時間であった．

実験では参加者に対して本システムの使い方を説明し，練習用に用意したフォーメーションや移動を本システムで作ってもらったのちに，図 9 に示す 2 種類の課題の再現と参加者自身が考えた自由な作品の制作を本システムを使って行ってもらった．自由制作については最低でも 10 人以上のダンサーが 2 回以上移動するような作品を作るよう依頼した．各作品制作は参加者が満足した時点で終わりとし，実験後にインタビューをしてシステムの感想を聞いた．

なお実験時の実装ではコスト関数 d に距離の 1 乗と 2 乗との 2 種類を用意し，生成する候補数を倍にしていたが，候補整理の処理が複雑な割に距離の 1 乗を使った場合の候補はほとんど提示されなかったため，本稿では 2 乗のコスト関数のみを基準として説明している．

5.2 実験結果

参加者が制作したフォーメーションの移動における各経路の数を表 1 に示す．表の中の「手描」はユーザが描いた経路,「提示」は本システムによって提示され，そのまま採用された経路であり,「修正」はシステムに提案された経路を採用したのち，ユーザが編集を加えた経路を指す．この表より，実験で作成された全経路のうち提示経路の割合が 63%を超えていたことから，本システムはユーザの望む経路をある程度予測，提示できていることが分かる．

表 1. 参加者の制作したフォーメーションの移動における各経路の内訳．

	手描	修正	提示	合計
経路数	71	14	149	234
割合	30.34%	5.98%	63.68%	100%

また，実験後のインタビューでは全ての参加者が本システムの候補提示機能について，移動経路を手で描く手間が省けて便利だと評価した．個別では,「候補がサムネイル形式で提示されるのが分かりやすい（P2)」,「複数の経路が提示されるので自分では思いつかないような移動経路も見つけられる（P4)」などの意見があった．一方で分かりやすさにおいては「移動前後の位置に振られた番号が移動の対応を表しているように見える（P2)」という意見があり，他の参加者もそのように誤解して操作しているように見えたため，改善が必要だと考えられる．提示された候補の精度については P1 と P2 は満足していたが，P3 と P4 は思い通りの結果を得られなかったと答えた．P3 はフォーメーションの右半分の移動経路を作成後にその構造を保ったまま左半分に経路をコピーするような候補の生成を望んでいた．

6 まとめと今後の課題

本研究では候補提示型インタフェースを備えた，ダンスのフォーメーションや移動を作成するためのシステムを提案した．提案システムはユーザが描いた一部のダンサーの移動経路をもとに残りのダンサーの経路を予測し，候補を生成して提示するため，振付師が移動経路を作成する手間と時間を節約することができる．現在の実装では候補の予測をルールベースで行っており，予測できる移動の種類や精度が不十分で改善の余地があると考えられる．ダンスのフォーメーションや移動のデータを大量に集めることができれば，機械学習の手法を使うことでよりユーザの意図に合った候補の生成，提示が可能になると考えており，今後の方向性として検討している．

謝辞

本研究は JST CREST JPMJCR17A1 の助成を受けたものである．

参考文献

[1] choreotechnology. Dance Designer Choreography Software（最終閲覧日：2021 年 9 月 12 日）. https://youtu.be/q4k4MAoaFZo, 2009.

[2] K. Endo, S.-T. Noh, K. Nakashima, X. Yang, and T. Igarashi. A Suggestive Interface for Designing Dance Formations. In *ACM SIGGRAPH 2021 Posters*, SIGGRAPH '21, New York, NY, USA, 2021. Association for Computing Machinery.

[3] Q. Gu and Z. Deng. Generating Freestyle Group Formations in Agent-Based Crowd Simulations. *IEEE Computer Graphics and Applications*, 33(1):20–31, 2013.

[4] J. Henry, H. P. H. Shum, and T. Komura. Interactive Formation Control in Complex Environments. *IEEE Transactions on Visualization and Computer Graphics*, 20(2):211–222, 2014.

[5] T. Igarashi and J. F. Hughes. A Suggestive Interface for 3D Drawing. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*, UIST '01, pp. 173–181, New York, NY, USA, 2001. Association for Computing Machinery.

[6] T. Kwon, K. H. Lee, J. Lee, and S. Takahashi. Group Motion Editing. *ACM Trans. Graph.*, 27(3):1–8, Aug. 2008.

[7] J.-s. Lee. An Intuitive Mobile Application for Notation of Group Dance Floor Plan. In C. Stephanidis ed., *HCI International 2014 - Posters' Extended Abstracts*, pp. 349–354, Cham, 2014. Springer International Publishing.

[8] J.-S. Lee and S.-H. Lee. Automatic path generation for group dance performance using a genetic algorithm. *Multimedia Tools and Applications*, 78(6):7517–7541, Mar. 2019.

[9] C. D. T. Mathew, B. Benes, and D. Aliaga. Interactive Inverse Spatio-Temporal Crowd Motion Design. In *Symposium on Interactive 3D Graphics and Games*, I3D '20, New York, NY, USA, 2020. Association for Computing Machinery.

[10] P. Merrell, E. Schkufza, Z. Li, M. Agrawala, and V. Koltun. Interactive Furniture Layout Using Interior Design Guidelines. *ACM Trans. Graph.*, 30(4), July 2011.

[11] Microsoft. Microsoft PowerPoint（最終閲覧日：2021 年 9 月 12 日）. https://www.microsoft.com/ja-jp/microsoft-365/powerpoint.

[12] J. Munkres. Algorithms for the Assignment and Transportation Problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, 1957.

[13] S. Murugappan, S. Sellamani, and K. Ramani. Towards Beautification of Freehand Sketches Using Suggestions. In *Proceedings of the 6th Eurographics Symposium on Sketch-Based Interfaces and Modeling*, SBIM '09, pp. 69–76, New York, NY, USA, 2009. Association for Computing Machinery.

[14] P. O'Donovan, A. Agarwala, and A. Hertzmann. DesignScape: Design with Interactive Layout Suggestions. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pp. 1221–1224, New York, NY, USA, 2015. Association for Computing Machinery.

[15] S. Patil, J. van den Berg, S. Curtis, M. C. Lin, and D. Manocha. Directing Crowd Simulations Using Navigation Fields. *IEEE Transactions on Visualization and Computer Graphics*, 17(2):244–254, Feb. 2011.

[16] A. Schulz, W. Matusik, and L. Velho. ChoreoGraphics: An Authoring Tool for Dance Shows. *Journal of Graphics Tools*, 17(4):159–176, 2013.

[17] A. Soga. and I. Yoshida. SIMULATION SYSTEM FOR DANCE GROUPS USING A GAMEPAD. In *Proceedings of the International Conference on Computer Graphics Theory and Applications and International Conference on Information Visualization Theory and Applications - Volume 1: GRAPP, (VISIGRAPP 2012)*, pp. 365–368. INSTICC, SciTePress, 2012.

[18] StageKeep. StageKeep（最終閲覧日：2021 年 9 月 12 日）. https://stagekeep.com.

[19] UGOKU. MO-SIM（最終閲覧日：2021 年 9 月 12 日）. https://mo-sim.com.

[20] B. Ulicny, P. d. H. Ciechomski, and D. Thalmann. Crowdbrush: Interactive Authoring of Real-Time Crowd Scenes. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '04, pp. 243–252, Goslar, DEU, 2004. Eurographics Association.