AUTOMATIC PATH GENERATION FOR GROUP DANCE PERFORMANCE USING A GENETIC ALGORITHM

Jeong-seob Lee **Sung-Hee Lee**

Graduate School of Culture Technology
Korea Advanced Institute of Science and Technology
291, Daehak-ro, Yuseong-gu, Daejeon, South Korea
jslee85, sunghee.lee@kaist.ac.kr
+10-42-350-2925

Abstract

Designing dancers' paths in choreography, or a floor pattern, is one of the most highly creative tasks of choreographers. Aiming to assist this task, this paper presents a novel system that automatically generates a number of floor patterns for multiple dancers given a choreographer's high-level feature inputs. The proposed floor pattern model represents locomotor movements of dancers on stage. Through a dance literature survey, four major features, i.e., time, space, symmetry, and entropy, were selected as feature inputs and mathematically modeled. Our system uses a multi-objective genetic algorithm to achieve desired floor patterns given input features. It iterates from random floor patterns to the ones that satisfy users' preferences while exploring the space of floor pattern with selection, mutation, crossover methods that are developed to fit the genotype of our system. User tests confirmed that our system generates a wide range of floor patterns according to user-specified input conditions. In addition, an actual dance piece was choreographed with the proposed method, which validated the usefulness of the proposed system. The proposed system is the first that automatically generates floor patterns for multiple dancers.

Keywords

Choreography; dance floor pattern; genetic algorithm.

Acknowledgements

This research was supported by the Korea Advanced Institute of Science and Technology (KAIST) and Daejeon Culture and Art Foundation.

1 Introduction

Dance is an art of human body and motions. A flourishing combination of postures and body gestures builds up an artistic experience for an audience. Various movements have been created to express mythical stories, symbolic figures, and emotions. Knowledge of physics and anatomy has expanded dance's boundaries.

In dance, the locomotor movements of dancers are an important element that makes dance more dramatic and sophisticated. In a dance piece, dancers unceasingly move around on stage as a part of choreographed dancing motions or for changing locations. The moving paths can have various forms. Multiple dancers may move as a group or get divided to several groups or to individuals. The dancers move fast or slow, straight or round. Their movements may be ordered or chaotic. Dancing is an exploration between stability and instability, as well as between balance and chaos. The relations or interactions among dancers make the paths more complicated. These varieties of floor patterns created by the locomotor movements are closely related to the theme, story, context, or mood of the scene, music, and other elements of a dance piece. Therefore, the design of the choreographic path, or floor pattern, is a complicated problem that requires consideration of many aspects.

Traditionally, the creation of choreography has relied only on the choreographers' knowledge and experience. It typically requires a huge time and effort to get inspiration and refine the design. Moreover, to achieve an actually implementable floor plan from an initial idea, dancers need to test and evaluate the plan repeatedly, which may be exhausting. While such a process is still prevalent in the field of dancing, some artists in dance, such as Merce Cunningham, have explored new methods to acquire new ideas. They have brought improvisation and external aleatory elements as alternative tools. Recently, researchers have been experimenting with computational approaches to stimulate the creativity of choreographers by digitizing, archiving, analyzing, synthesizing, and displaying dances for educational or creative goals [1–4]. Our research shares the same philosophy as them.

Our research aims to build a computational system that assists the creative process of designing a dance floor plan. Given several target features of the floor pattern by a choreographer as inputs, our system generates numerous patterns that satisfy the conditions. From an engineering perspective, choreography is an exploration of the high-dimensional space of human movements to find novel solutions. As choreography is about artistic and creative inspiration, it tends to have multiple objectives and encourages diverse solutions. For these reasons, we selected a multi-objective genetic algorithm as the main frame of our system. The artistic background for the system came from studies of choreography theory. From countless elements to consider for choreography, we chose a small number of important features as a foundation of our work. For example, Rudolf von Laban's theory of effort [5] and symmetry played a major role in our work. Selected features were quantitatively modeled to evaluate candidate floor patterns. With this setup, our system based on the evolutionary algorithm explores the space of floor patterns by randomly generating sample floor patterns, and gradually generates solutions that satisfy the choreographer's desires. We performed a user test to evaluate our system, and the results show that there is a significant positive correlation between the intended input features given by a choreographer and the perceived features by users.

Contributions. In previous decades, many artists have experimented with collaboration of dance and other media on stage. However, it is only relatively recent that they started to change dance more fundamentally with computational means, as will be discussed in the next section. To the best of our knowledge, our system is the first that automatically generates floor patterns for multiple dancers, and this is in contrast with existing exploratory work that focused mostly on the motion of individual dancers. In terms of practicality, our system can be used for dancers and choreographers in the field. Regarding technical contributions, we show that a genetic algorithm-based framework can generate plausible floor patterns for dancers. This is enabled by our novel methods to quantitatively apply choreography theories of Laban and others to non-human shape structure.

The rest of this paper is organized as follows: In Section 2, we review some academic studies related to our work. The contextual background of our topic and technical methodology that we used are also introduced. Details of our system are described in Section 3. The user scenario, structure of our system, floor pattern model, and feature models to evaluate floor patterns are also detailed in this section. The user evaluation of the system and its results are reported in Section 4 and Section 5 concludes the paper.

2 Related work

2.1 Efforts for artistic creativity

In the long history of art, acquiring new ideas for creation or inspiration has always been important to artists. Choreographers also have searched for their own source or methodology for the inspiration. The movements in dancing should be driven by reasons or motivation. Music has been the most basic element that leads dancers rhythmically and emotionally. Literature has also been another popular source to motivate dancers. Physical objects like a ball contributes in narrowing down possibilities of progress in movement.[a]

More lately, improvisation has become one of the most popular techniques for contemporary choreographers. Without a predetermined choreography, dancers or choreographers decide the next moves instantly and intuitively [6]. Some abstract clues, such as music, situation, language, and physical items are used to help. As this decision process, one can reach motion space beyond conscious boundaries. The improvisation technique lets the dancer and choreographer explore infinite possibilities in the space of dancing.

Historic choreographer Merce Cunningham and musician John Cage once used *I Ching*, the Chinese book of changes, or Hexagram, to form their own choreography and music for it. This kind of aleatory technique enables artists to adopt inspiration from the outside world [7, 8].

[a] Akiko Takeshita, YCAM InterLab, Kyoto Experiment Talk, 2017.

These improvisation and aleatory methods have characteristics of a stochastic process, which notably has a lot in common with the solution seeking process of evolutionary algorithms.

2.2 Computational motion design

More sophisticated approaches to choreograph with computational means have been found in the last decade. Gentry et al. [3, 9] noticed that swing dance steps are well standardized and can be coded with a small number of movement vocabularies. By taking advantage of these characteristics, they modeled the entire swing dance as a finite state machine with the movement vocabularies as the states and automatically generated dance sequences.

Similarly, LaViers et al. [2, 10] noticed that the basic leg postures of ballet are highly standardized. They sampled ten leg postures from basic ballet and developed a system that generates ballet sequences by transitioning between the sample postures. Lee et al. [11] investigated the musical aspects of dance. Their work generated choreography by adjoining short motion segments based on the correlation between music and motion segments. Pathfinder[b] is another example of a generative system that provides dancers with other possibilities of motions by showing abstract animated geometrical figures that constantly deform. In this way, there have been various computational approaches of choreography, which are mainly derived from the robotics or motion synthesis fields for computer graphics.

Laban's motion analysis (LMA) theory is frequently used in this context. Rudolf von Laban, the pioneer of dance motion analysis, introduced effort theory, which defines four dimensions, namely, time, space, weight and flow, to describe the fundamental dynamic characteristics of movement [5]. His work enabled dancers to think of body movement more objectively and scientifically. While it was basically a tool for dance education, creation, and research, it expanded its application to sports, general movements, and robotics as well. Yet, these are descriptive, conceptual dimensions, and thus many researchers have proposed mathematical formulations for their own purposes [12–14]. In this paper, we applied LMA theory to identifying features to evaluate generated floor patterns.

2.3 Beyond aleatorism – Evolutionary art

Genetic algorithms have made flourishing contributions in various areas to solve complicated high dimensional problems [15]. While mostly used for practical engineering problems, the genetic algorithms have also been used for artistic purposes [16–18]. Besides the ability to find optimal solutions, their stochastic nature is a very appealing feature to come up with novel ideas [19]. Artistic decision-making tends to have multiple heterogeneous goals and thus requires a multi-objective approach. Pareto optimality [20], which we used in our work, is one of the most popular methods for this multi-objective problem.

For choreography, Carlson et al. [1] used genetic algorithms for contemporary choreography. Their concept of movement catalyst aims to encourage choreographers to distance themselves from their comfort zone of choreographic habits and to explore novel movements. Likewise, Lapointe and

---

[b] http://princemio.net/portfolio/pathfinder/

Époque [4] introduced an interactive genetic algorithm with motion capture for computer-aided choreography. We also apply genetic algorithms to choreography. However, unlike other work [3, 17] that focused on the motion of an individual dancer, our main interest is generating paths of multiple dancers, which is another crucial element of choreography. To this end, we developed a floor plan model and a set of fitness functions that evaluate certain high-level characteristics of paths for multiple dancers.

Crowd simulation research in computer graphics field has developed a number of methods to model crowd configurations, which can benefit choreography of multiple dancers. Kwon et al. [21] and Henry et al. [22] treated crowds as a mesh structure, where they converted 2D paths of multiple agents to a 3D mesh by connecting agent positions at the same time with edges. We adopt this idea to represent dancers' positions in our research. Johansson and Helbing [23] introduced pedestrian simulation with a genetic algorithm for the indoor architectural design.

3 Proposed System

3.1 Overview

The overall structure of our system is shown in Fig. 1. Firstly, a choreographer would have a feature preference about what he/she wants to create. Through a survey on choreography theory, we collected numerous elements that are related to choreography and selected 4 fundamental features: time, space, symmetry, and orderliness. Time and space were brought from Laban's effort theory [5], which is one of the most recognized theories of motion analysis. Also, authoritative choreographers, like Doris Humphrey, stress symmetry and orderliness [6, 24]. As these features are descriptive, mathematical models to quantify each of them were developed. Based on these models, floor patterns can be evaluated and compared to the user's input preferences quantitatively. Meanwhile, our floor pattern model was designed to express the locomotor movement behaviors of dancers and be used as a genotype of our genetic algorithm. This is constrained by conditions such as the size of the stage and musical structure (i.e. tempo and metre). With this floor pattern model and feature models, our system runs a multi-objective genetic algorithm. One floor pattern is a sample solution of our genetic algorithm. It iterates from random initial floor patterns to the ones that satisfies the user's preference.

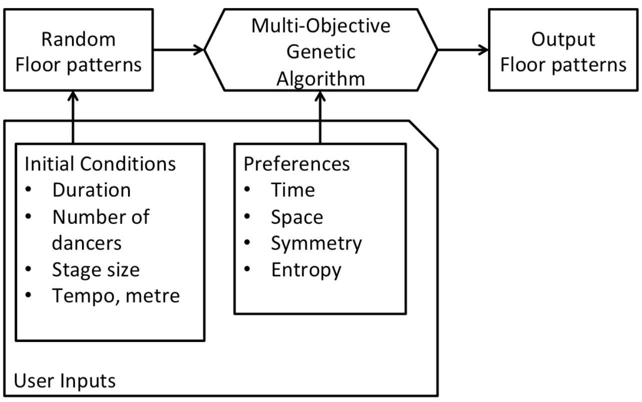


Fig. 1 Overview of our floor pattern auto-generation system

3.2 Floor pattern model

We developed our floor pattern model to express the behaviors of dancers, and it consists of a number of elements, which are described in detail here.

The first element is the stage. The width and depth of the stage constrain the boundary of dancers' locomotor movements. We set them as the initial input parameters of the system.

Music is also an important element of choreography. The style and structure of music are strongly coupled with the movements of dancers [25]. The relation between music and dance movements is quite delicate. A faster tempo encourages dancers to move faster. Musicians often regard proper tempo for dancing as too slow and boring, and may think the performance gets better with varying tempos while dancers prefer a constant tempo [25]. Also, number of beats per measure, or metre, considerably changes the condition. Quadruple and triple metres demand a different combination of motions, which evoke different emotions. Instability and sensitivity of odd number of beats per segment is typical characteristic of triple metre compared to the quadruple metres. To reflect these basic musical features, we set tempo and metre as initial conditions of our system. The number of dancers is also set as an initial condition.

For the expression of dancers' behavior on the stage we surveyed dance notation systems. Laban's dance notation system, or Labanotation, is a well-known example. Labanotation is another representative work of Laban for expressing dancer's full-body motions scientifically. In Labanotation, floor pattern of dancers is basically visualized with pins and arrows on top view floor plan. Pins indicate dancers. Shape of pins differentiate type of dancers (e.g. gender and age). Arrow indicates path from a starting point to the end. The arrow(path) might be straight or curved. Coordination is musical score was also designed and encouraged. Relationship between dancers is noted with the shape of pins [26]. Many choreographers tend to draw their designed floor plan is a somewhat similar manner to Labanotation.

We modeled 4 types of locomotor movement behaviors of dancers from the examples of the Labanotation: translation, rotation, staying, and mirroring. Translation is the most basic element for linear displacement from one position to another. Rotation was added to reflect curved motion around a certain position. The destination positions of rotation and staying are decided relatively with their departing positions. As non-moving is as important as moving in choreography, we added staying behavior explicitly. A relationship is a keyword that stands out in dance compared to other art genres. Confronting, cooperating or other kind of relationships between dancers create various situations on stage. For this reason, one noticeable characteristic of the project Motion Bank[c] is archiving the relationship between dancers on stage. To reflect one of such relational aspects, we added a mirroring key that allows for a dancer to move with respect to another dancer. Mirroring key refers to a reference dancer and copies his/her displacement path (i.e., path relative to the departing position) in a certain time duration. It can be copied as it is, or be flipped horizontally or vertically (Fig. 2).

---

[c] http://www.motionbank.org

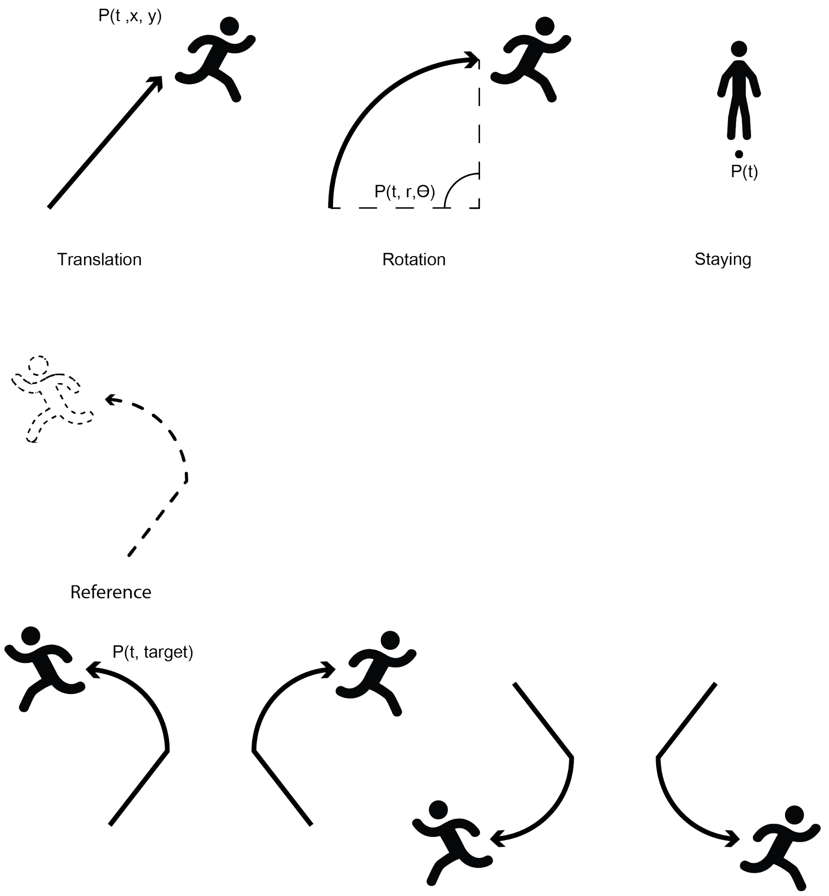
A path generated by the above locomotor movement behaviors are represented by a sequence of key values at key frames. The key values for the four behaviors specify parameters for the behavior, as defined as follows:

- Translation: end time, destination coordinates
- Rotation: time, center of rotation, rotation angle
- Staying: time
- Mirroring: time, target dancer, mirroring style (copy, vertical flip, or horizontal flip)

The sequence of these motion keys define a timeline for each dancer about when and how they behave.

Locomotion of dancers in choreography is rather planned than pedestrian in usual situations. For this reason, a path in choreography shows longer geometric strokes than that of general walking motions. Therefore, we set the length of one musical measure as time intervals between the key frames of a dancer’s path. It is different from common representation of locomotion used in computer animation (e.g. array of irregular coordinates per tens of milliseconds).

Overall, our floor pattern model can be represented with a two-dimensional matrix (Fig. 3), where each row and column denotes the identity of a dancer and the key frame, respectively. Each element in the matrix specifies the key values of one of the four behaviors as visualized in different colors.



Mirror (with four flip combinations against the reference)

Fig. 2 Four types of motion keys[d]

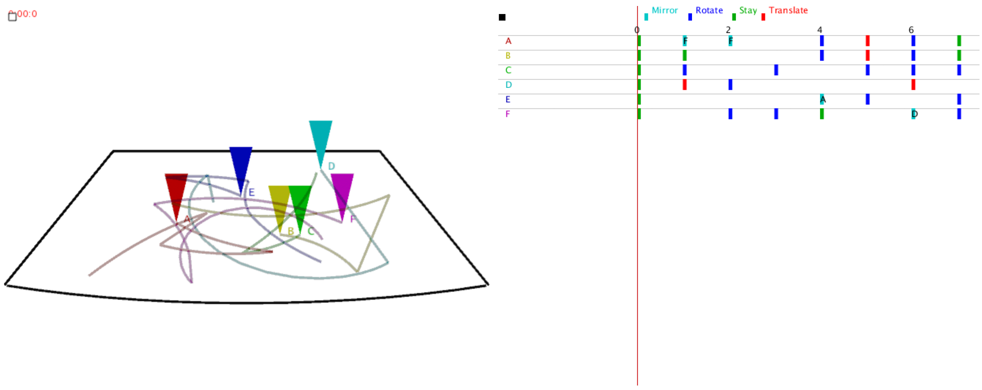


Fig. 3 Visualization of our floor pattern and its genotype in form of time line (Triangles indicate dancers)

3.3 Features for solution evaluation

(To reviewers: please refer to video clips that contains some samples.)

After surveying studies on choreography theory, we determined features to evaluate floor patterns. We started with the four features of Laban's motion analysis theory: time, space, weight, and flow. Weight and flow are rather abstract and composite features to express how the motions look. Weight element classifies movements with strong and light. Likewise, flow element has two elements: bound and free movement. As they are abstract and composite, further study is required to apply them to floor plans. In our work, we adopted only time and space elements. Symmetry is another key aspect of choreography. Doris Humphrey, dancer and choreographer of early contemporary dance, proposed symmetry as the first constant for the design of dance [24]. Lastly, we measured the entropy of a floor plan to evaluate the group-wise behavior of dancers. Each feature is explained next.

3.3.1 Time (Speed)

As aforementioned, time is one of four parts of Laban's theory and has two elements: sudden and sustained. With respect to floor pattern design, time can be related to the speed of locomotion. The audience feels more dynamics with a faster motion. To evaluate this aspect, we select the average speed v_{mean} of dancers as the first kinetic feature element and define a cost function to minimize its difference from the desired average speed. Fig. 4 shows examples of two floor patterns obtained with low and high desired average speeds, respectively.

$$cost_{time} = |v_{mean} - v_{mean,goal}|$$

[d] Figure contains images created by Vladimir Belochkin and Mourad Mokrane from the Noun Project.

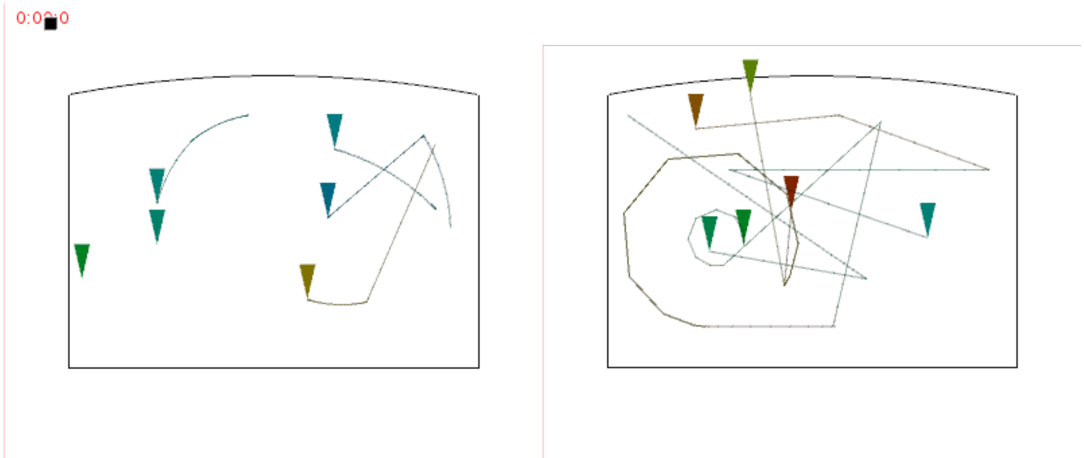


Fig. 4 Top view of generated floor patterns with different desired average speed (0.23m/s for the left, 1.10m/s for the right. Other features were controlled to be nearly equal.) Colored triangles and strokes indicate the current position and paths of dancers

3.3.2 Space (Direct or curved)

Another element of Laban's effort theory is space, which divides movements into direct or indirect, straight or curved [5]. Blom and Chaplin [6] commented that "curved design and straight-and-angular design show a marked and distinctive contrast in their dramatic implications". Straight or rounded line movements evoke different impressions such as aggressiveness and elegance. LaViers and Egerstedt [12], who used Laban's effort theory to synthesize the robot's motion with style, implemented their own method to quantify this quality. They modeled the space dimension as the distance between the linear interpolation of two adjacent key frames and an actual path. However, this method may recognize a sharp zigzag path as a curved one. Instead, we computed the average of the curvatures for every three consecutive frames in each path to measure the space quality. The curvature was estimated with Menger curvature.

$$c_{mean} = \frac{\sum_i^N \sum_t^T c\left(p_{i,t-1}, p_{i,t}, p_{i,t+1}\right)}{N \times T}$$

$$cost_{space} = \left|c_{mean} - c_{mean,goal}\right|$$

where c denotes Menger curvature computed from three consecutive positions $p_{i,t}$ of a dancer i at time frame t. The variables N and T denote the number of dancers and the total number of time frames, respectively. Fig. 5 shows examples of two floor patterns with the desired value of space element set for straight and curved paths, respectively.

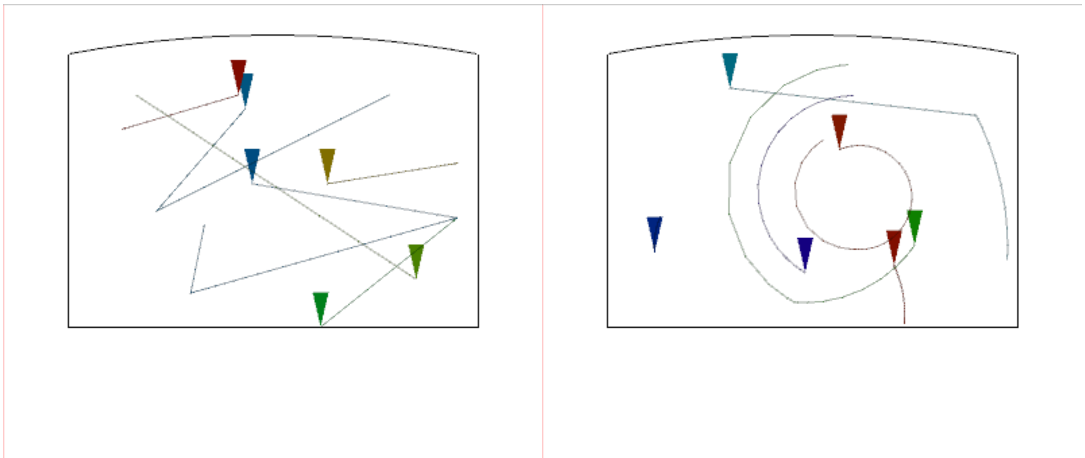


Fig. 5 Top view of generated floor patterns with different desired space values (0 for the left, 0.328 for the right. Other features were controlled to be nearly equal)

3.3.3 Symmetry

In general, symmetry suggests stability, balance, and security [6, 24]. However, in the world of aesthetics, the virtue of symmetry sometimes is regarded as a weak point because art is for stimulation, excitement, and adventure rather than comfort. Asymmetry stimulates those senses. The unpredictability of contests and imbalance is a typical impression of asymmetry for the audience. Thus, a proper balance between symmetry and asymmetry is needed by choreographers [6, 24]. Humphrey [24] stated that the design of dance is handling a complicated structure with two aspects (space and time), and there is symmetry in time as well as symmetry in space [6].

For measuring symmetry of dancers' movement, we adopt the method of Mitra et al. [27] which identifies symmetric geometries from an image or 3D geometry. To measure symmetry, [27] utilized the iterative closest point (ICP) method, a common method for finding the best-fitting transformation of images [28, 29]. ICP works by minimizing the distance between a transformed point cloud and an original one, and the distance, or ICP residual, can be a measure for symmetry [27].

Basic principle of the ICP method is as follows [30]. Let us assume two point clouds $P = \{p_1, p_2, \cdots, p_n\}$ and $Q = \{q_1, q_2, \cdots, q_m\}$, and Q is transformed by a transformation α. First step is to identify a nearest point $\alpha(q_j)$ among $j = 1 \dots m$ to each point p_i. Then, sum of the squared distance between the pairs ε(α) is called ICP residual error

$$\varepsilon(\alpha, P, Q) = \sum_{i=1}^{n} d\big(p_i, \alpha(q_j)\big)$$

where $d(\cdot,\cdot)$ denotes a distance between two points. In our experiment, we use the Euclidean distance.

Obtaining an optimal transformation α that minimizes the residual is the goal of ICP method. In our work, the residual is used to evaluate symmetry of floor patterns. How can we utilize this method to our motion data? And how can we reflect the temporal aspect of symmetry? For this, we used the approach of Kwon et al [21]. They modeled the locomotion paths as a 3D mesh in which the third axis represents time. Likewise, we converted our floor pattern as a 3D point cloud with time as the third axis (Fig. 6). In this way, we can include temporal aspects in addition to the spatial symmetry for considering symmetry. For instance, symmetrical paths with a slight time delay can also be detected.

In our work, we only consider symmetry of dancers' paths with respect to the vertical center axis. Therefore, our cost function for symmetry is

$$cost_{symmetry} = |\varepsilon(r_v, P, P) - \varepsilon_{goal}|$$

where r_v denotes reflection transformation in the vertical center axis and P is the point cloud made by dancers' paths. The smaller $\varepsilon(r_v, P, P)$ is, the more symmetrical the floor pattern is. Fig. 7 shows examples of two floor patterns with high and low symmetry values.

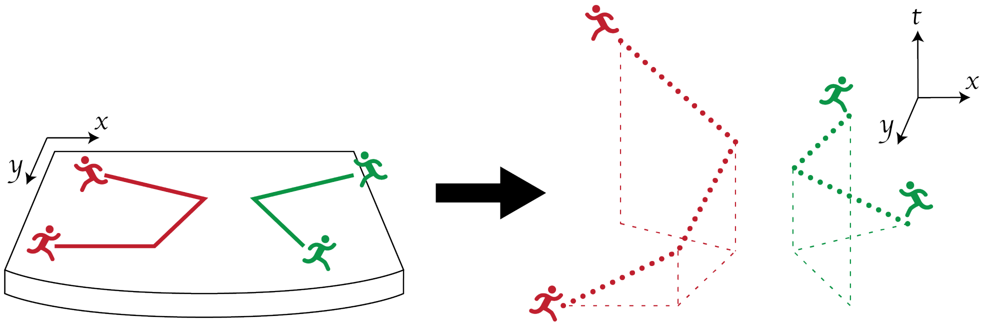


Fig. 6 Conversion from 2D paths (left) to 3D point clouds (right). Vertical axis represents time

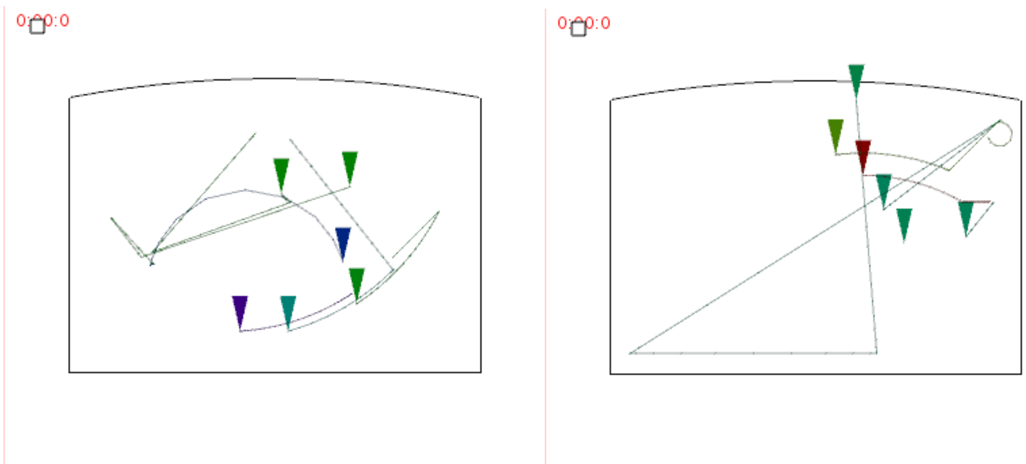


Fig. 7 Top view of generated floor patterns with different desired symmetry values (ICP residuals are 0.611m for the left, 4.179m for the right. More symmetric result is obtained in the left example. Other features were controlled to be nearly equal)

3.3.4 Entropy

The three features introduced so far are related to individual paths and do not control group-wise behavior. As we mentioned in the beginning, the group-wise behavior is an important element of dance piece. Specifically, we focus on controlling the uniformity or complexity of group motion. To achieve this property, we used the concept of entropy. Shannon's entropy is a measure of information. Eduardo et al. [31] used this concept to measure the diversity of solutions in their optimization method. Oulasvirta et al. [32] analyzed the full-body motions of a dancer with information capacity, or throughput. In our work, the entropy is used for measuring the degree of complexity of the group-wise behavior of our floor pattern.

At each time frame, the angle of movement direction of all dancers are collected and the histogram of the angles is calculated. For staying dancers, they are categorized separately in the histogram as it is not possible to estimate their direction. The discrete probability distribution of the angles is then obtained by dividing the histogram by the number of dancers. Entropy is then estimated from the angle probability histogram.

$$\mathrm{H}_{mean} = \frac{\sum_t^T \sum_i^{N_{bin}} \left(K \times Prob_{t,i} \times \log Prob_{t,i}\right)}{T}$$

$$K = -\log(N_{bin})$$

where $Prob_{t,i}$ denotes the possibility of the i'th direction angle at time t and N_{bin} is the number of histogram bins. We divided the angles into 21 bins where 20 bins for each angle and 1 for zero velocity.

The entropy at a time step is calculated to be low if the dancers move in the same direction at the moment. Conversely, the entropy is higher when the dancers move more individually, randomly, or chaotically. The average of the entropy over every time frame is used as the cost function. Fig. 8 shows example of two floor patterns with low and high entropies.

$$cost_{entropy} = \left|\mathrm{H}_{mean} - \mathrm{H}_{mean,goal}\right|$$

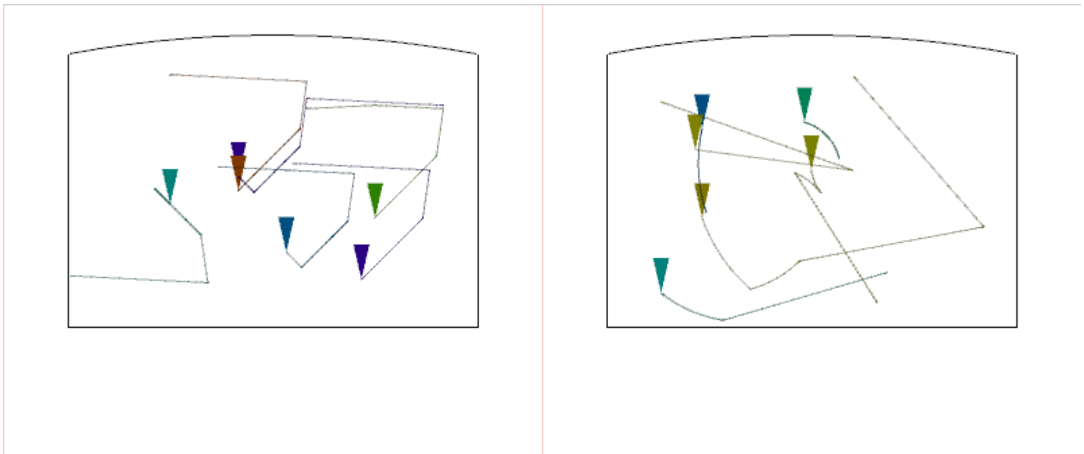


Fig. 8 Top view of generated floor patterns with different entropy values (0.012 for the left, 0.517 for the right. Other features were controlled to be nearly equal)

3.4 Evolution

With the floor pattern and feature models described above, our system runs a genetic algorithm to obtain a floor pattern with the desired properties. Specifically, a multi-objective genetic algorithm is used. To this end, one needs to set initial conditions, which include the size of stage, musical structure, and the number of dancers. Additionally, one sets the desired values of the features, which are time, space, symmetry, and entropy.

Based on the initial conditions, the first generation with random genotypes is generated. Subsequent generations are created with a conventional process of genetic algorithms: selection, crossover, and mutation. One cycle of these three processes yields two new solutions. After testing physical feasibility of each solution, adequate ones are added to the set of new generation. The selection, crossover and mutation process are repeated until it fills the set of new generation. Pseudo code of this process is in Fig. 9 and details of each process will be described below.

```
var solcount=0
     newGeneration = {∅}

     While solcount < 400{

       sol = randomly select two solutions from the last
       generation

       crossover sol[0] and sol[1]

       for(i=0 ; i<2 ; ++i){
           mutate sol[i]

             if sol[i] is feasible & solcount<400
               Add sol[i] to newGeneration
               solcount++
       }
     }

Return new newGeneration
```

Fig. 9 Pseudo code for generating one set of new solutions

Selection. Pareto ranks [20] of each solution in the latest generation are obtained and those ranks are converted into fitness. Pareto dominance is suited for comparing elements with multiple features in different units and scales. Its basic idea is introduced in Fig. 10. By comparing each feature respectively, one solution is called dominant to another if it is superior in all aspects. Pareto rank of one solution is the number of other solutions being dominated by the solution. The Pareto ranks are converted to fitnesses to run roulette-wheel selection. The formula for the fitness of the i'th solution is as follows:

$$fitness_i = \frac{rank_i - rank_{min}}{rank_{max} - rank_{min}} s + (1 - s)$$

where $rank_i$ is the rank of the i'th solution, $rank_{max}$ and $rank_{min}$ are maximum and minimum ranks in all solutions. The parameter s denotes the selection pressure, which decides the selection chance gap between the best and worst solutions. With $s = 0.8$, the fitness guarantees a minimum 20% of selection chance for inferior solutions to encourage the diversity of solutions.

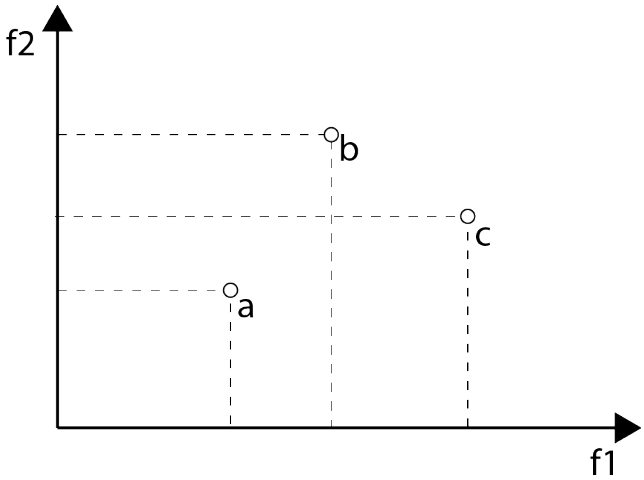


Fig. 10 Example of Pareto dominance in 2 feature (f1 and f2) space. Assuming the minimum-cost problem, a is lower in all features than b and c. It makes a dominant to both. Solutions b and c are superior in one feature and inferior in the other, so they are not dominant to each other

As another means for promoting diversity, we applied the fitness sharing and niching method [33]. The fitness sharing technique keeps the population diverse by penalizing the fitness of solutions if there are similar solutions in the same generation. Niching considers the closeness of two solutions for the penalty. The similarity of solutions is estimated with the ICP residual between two solutions, which was used for symmetry. For each solution, its niche count, which is the number of other solution in range of its niche, is calculated and its fitness is divided by its niche count as follows:

$$fitness_i \leftarrow \frac{fitness_i}{niche\ count_i}$$

Among the whole number of solutions in each generation (mostly 400 in our work), two solutions are selected with the roulette-wheel method [15] based on the estimated fitness. With these two solutions, crossover process follows.

Crossover. Normally, the crossover method differs depending on the structure of the chromosome and the goal of computation. The chromosome of our work is represented as a 2D matrix. Rows and columns represent the dancers and time respectively, and elements of the matrix are motion keys. Therefore, a certain area of the matrix decides the behavior of some dancers within some time range. Therefore, cutting off and swapping the arbitrary areas of the matrix is a reasonable approach for the crossover. We chose rectangle as the shape of the crossover window [34]. Fig. 11 is an example of our crossover method.

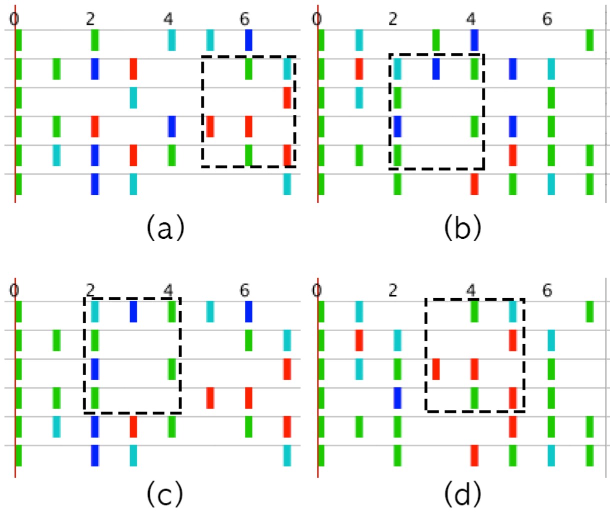


Fig. 11 An example of crossover operation. Square areas of (a) and (b) are swapped to generate child solutions (c) and (d)

Mutation. After the crossover, the mutation process follows. Though our chromosome structure is a 2D matrix, each element represents varying types of keys with different number of variables, which makes mutation trickier than typical genetic algorithms with binary or integer genotypes. In our work, we applied three kinds of stochastic mutations: removal, addition, or modification. For every row and column, i.e. every dancer and interval, suitable mutation operations were applied. Firstly, for the existing keys, each of them is removed with a certain probability. Secondly, for empty slots where no key is defined, new keys are added with a given possibility. The choice of key type and attributes are set randomly. Thirdly, the attributes of existing keys, including time and target dancer, are randomly changed with a certain probability within some range. For all those mutation operations, the mutation probability and the range of action are initially set heuristically and then gradually reduced with the evolution [35]. In the early stage of evolution, a higher mutation rate is recommended for the broader exploration of the solution, and it is lowered at a later stage to keep the discovered superior solutions. The equation for the mutation rate is as follows:

$$mutation\, rate = 1 - r^{(1-\frac{m}{M})^b}$$

where $r \in [0,1]$ denotes a random probability that changes every time, M is the maximum number of generation, m is the current generation, and b is a predefined constant, which was set to 5 according to Michalewicz's recommendation [35].

When a new solution is generated, we first examine its feasibility: Solutions that deviate from the stage boundaries or violate the speed limit of dancers are discarded before evaluating their fitness. At this point, one new solution of floor pattern is complete. This process is repeated until it reaches the maximum population. As aforementioned, pseudo code of the cycle is as Fig. 9. This whole process is repeated for every generations. Instead of implementing algorithm to monitor convergence of the solutions, we simply fixed the number of generations to 400, which were enough for convergence for most cases.

Fig. 12 shows a typical of convergence behavior of each cost term in our evolution system. This plot shows that our system successfully converges to a minimum cost, getting closer to the desired input parameters. Entropy increases from zero in the beginning because the initial floor pattern is a simple uniform collective movement.

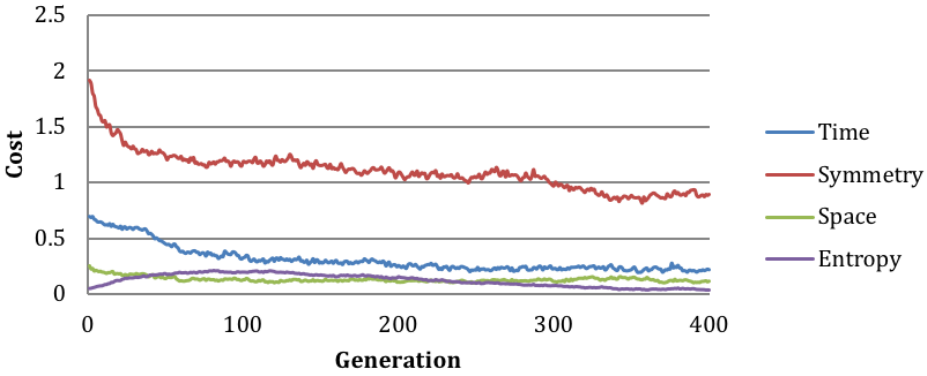


Fig. 12 An example of the time, space, symmetry, entropy costs change in evolution process

Fig. 13 shows a set of generated floor patterns. We implemented a graphic user interface to browse generated results. Each floor pattern is visualized as a thumbnail to allow for easy browsing. Users can choose a floor pattern and view it animated on an extra viewer. Top row shows the first random floor patterns. Iteration proceeds from top to the bottom. Solutions in each row are arranged in decreasing order of the score. As it is inefficient to observe all 400 solutions in every 400 generation, the browser shows floor patterns in every 40 generations and only 10 best solutions in each generation. From the top row to the bottom, one can see that the initial random solutions evolve to various floor patterns that share common feature characteristics. In this example, floor patterns commonly show dynamic, straight and ordered style.

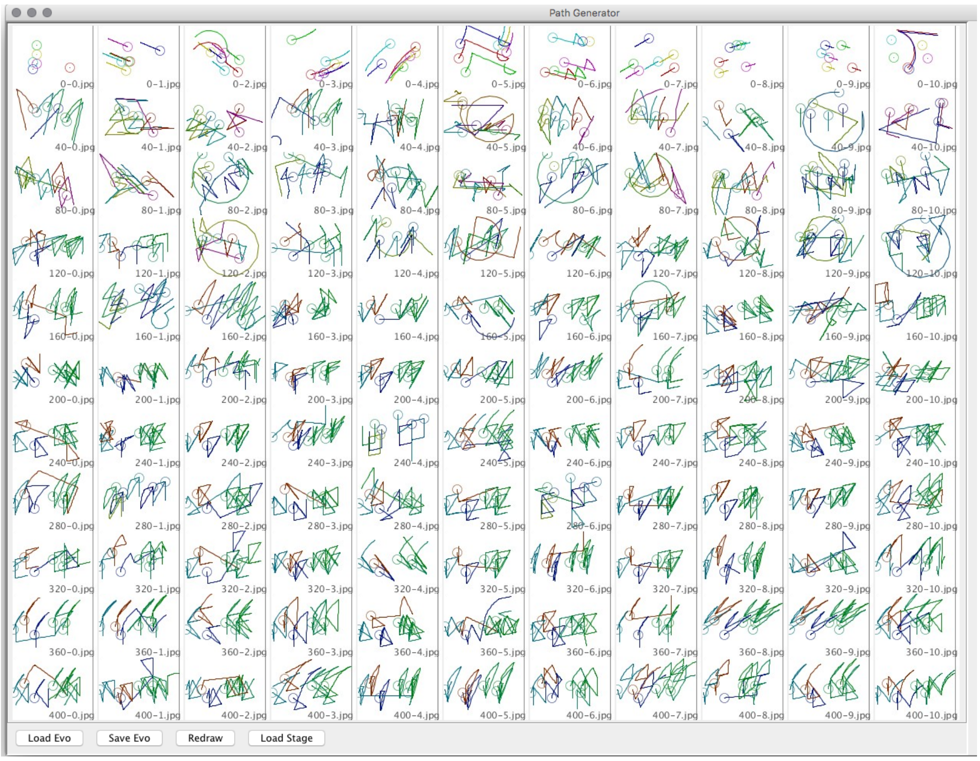


Fig. 13 Graphic user interface for browsing generated floor patterns. Initial random solutions and final optimized solutions are shown in the top and bottom rows, respectively

Our system was implemented with Java language and tested with a laptop PC equipped with a 2.8 GHz Intel Core i7 processor and 16GB 1600 MHz DDR3 memory. We observed that the solutions converge stably within 400 generations of iteration with constant 400 populations, which takes 22.6 minutes.

4 Evaluation

4.1 User test

In the previous section, we showed that our system successfully converges into solutions that satisfy user-given preferences. To evaluate whether the results generated with some desired feature values agree with an audience's perception, we performed a user test as follows.

Firstly, we generated 10 floor pattern sets with different feature values. Each of the four features has lower and upper bounds. These bounds are either due to the nature of the features (e.g., entropy cannot be lower than 0 and the ICP residual is constrained by the stage size) or some reasonable limit. The 10 sets of floor patterns were generated within these boundaries. Initial conditions set to the evaluation were as follows: Number of dancers: 6, duration: 15 seconds, BPM: 120, and number of meters: 4. The first set was the baseline in which every parameter was to the middle of the boundaries. Next eight sets were achieved by setting only one of the four features to either the maximum or minimum while fixing all other feature values to the median. As the last set, we created one popular scenario that has high velocity and symmetry and low entropy. For each of the 10 sets of feature values, we sampled two floor patterns with the lowest cost. Therefore, 20 floor patterns were sampled and these were used for the user test.

Test participants were shown video clips (Fig. 14) of each floor pattern. After watching each video, subjects were asked to answer four questions for each video (Table 1). Each question showed qualitative keywords for each feature and the subjects were asked to grade the floor pattern in five levels based on how they felt about the video. Subjects repeated this process for all 20 video clips. A total of 22 subjects participated in the test, among which seven had experience as a dancer on stage.

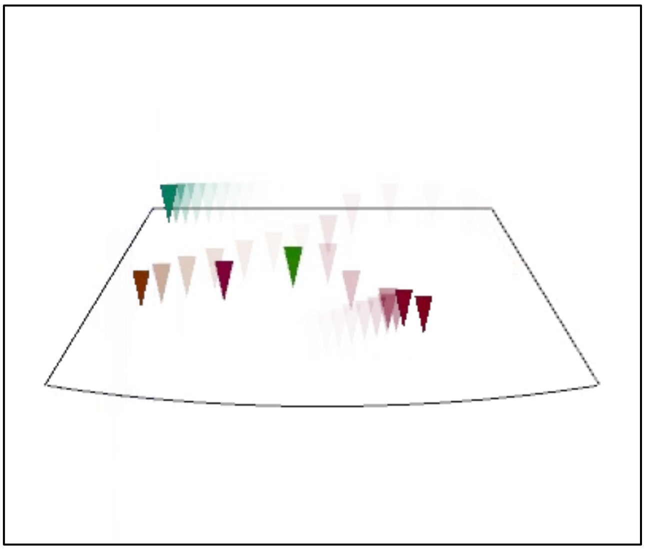
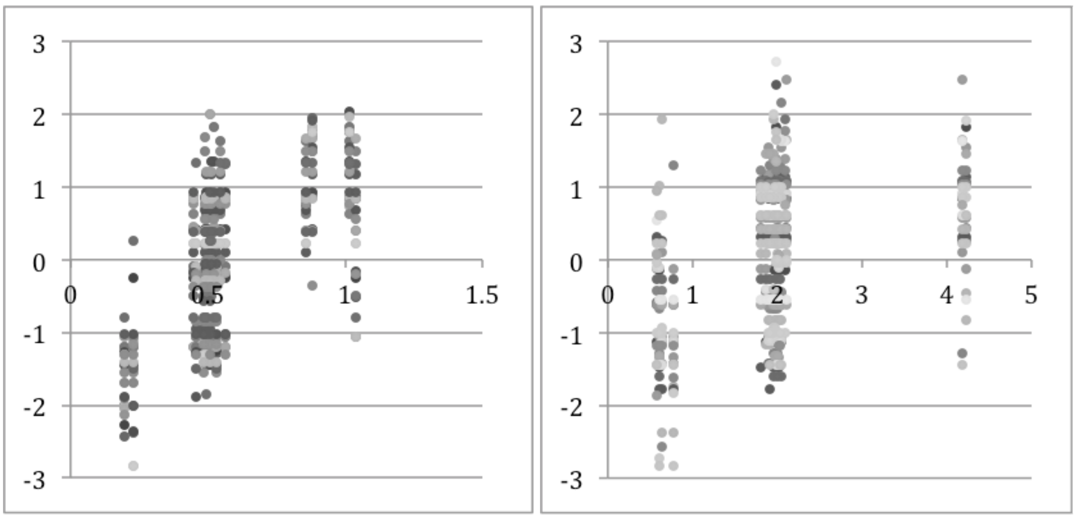


Fig. 14 Example of the floor pattern video clip

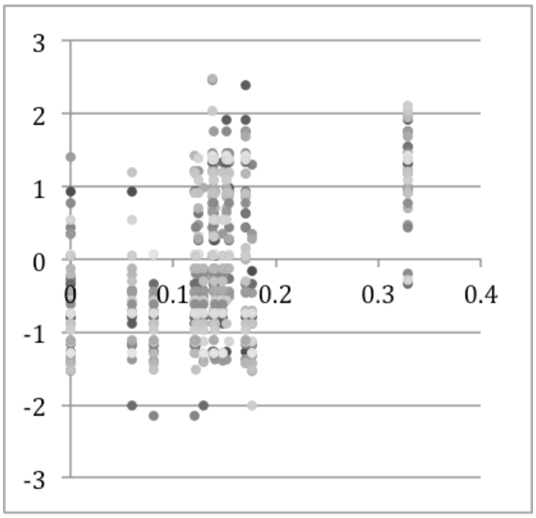
	1	2	3	4	5	
Slow, static						Fast, dynamic
Symmetry						Asymmetry
Straight, sharp						Curve, smooth
Ordered						Chaotic

Table 1 Questionnaire of the user test (translated from Korean). Subjects were requested to grade each row from 1 to 5 after watching each video clip

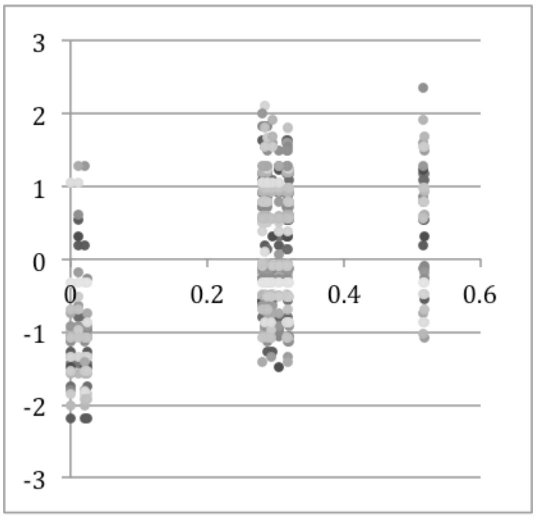
The test results were analyzed to find correlations between the subjects' responses and the actual feature values of each floor pattern. Firstly, to eliminate personal tendency in grading, the responses were normalized for each feature and each subject. The normalized responses and feature values were plotted in Fig. 15. The Pearson correlation of the normalized grades and actual feature values of the sample floor patterns were estimated. Higher correlation coefficient implies that, for each floor pattern, subjects perceived the intended intensity of feature values. Firstly, we ran the analysis for each subject and each feature separately, which led to 88 (22 by 4) coefficients. It showed promising results with the average of 0.488, which indicates moderate correlation. By summing up the data, the Pearson correlations of entire subjects for each feature were obtained and Table 2 shows the results. These correlation ratios imply moderate or stronger positive correlations, which means that the subjects recognized different impressions as intended when the feature values changed. The time (velocity) feature, the simplest feature, shows the highest score, and other features show similar scores while symmetry tends to be lower than others. There was no noticeable difference in observation between dancer and non-dancer groups (the second and third rows of Table 2).



(a) (b)



(c)



(d)

Fig. 15 Normalized responses (y-axis) and feature values (x-axis) of each feature of floor patterns. (a), (b), (c) and (d) represent the results of time, symmetry, space, and entropy, respectively. Different color indicates different participant

	Time	Symmetry	Space	Entropy
All subjects	0.604	0.405	0.461	0.485
Dancers	0.636	0.485	0.478	0.411
Non-Dancers	0.588	0.368	0.453	0.519

Table 2 Pearson correlations of each feature variable for each subject group

4.2 Demonstration

In the early stage of our research, we produced a dance piece by using our system. Although the early version of our system had some differences from the current system, it demonstrated how our system could be utilized in actual dance productions. The early system used the average speed and the average position of each path to measure the time and symmetry elements. Instead of the multi-objective algorithm, we used the linear combination of cost functions.

We produced a two-minute piece. One of the authors choreographed the first 50 seconds in a conventional way and then applied our system in the remaining 70 seconds. The paths for 70 seconds were generated and appropriate dancing motions were choreographed on the generated paths. Group-wise dancing motions were assigned to a group-wise path (Fig. 16). If a path is individual, it naturally became a solo part. When two dancers almost collided once, it was interpreted as a duet lift motion. Some segments in the paths were somewhat too slow, and such parts were utilized for theatrical expression rather than dancing.

If a choreographer designs such a complicated sequence in a conventional way, not only does it take enormous efforts to plan, but there are also problems with validating, recording, and sharing it with dancers. Our system showed its possibility as an efficient interface to overcome these problems.

Notably, throughout the choreography process, the floor pattern was actively interpreted and modified by a choreographer rather than being followed passively. In this context, the generated floor pattern serves as a clue or blueprint of the piece rather than a precise and absolute drawing, which agrees with the philosophy of Scuddle [1].

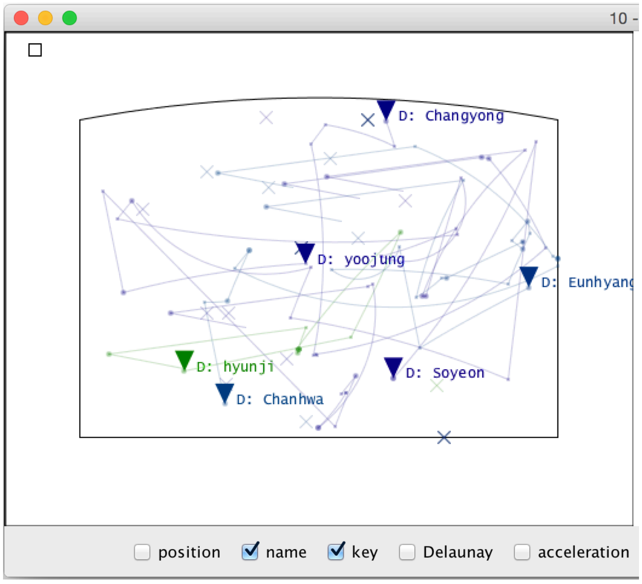


Fig. 16 Floor pattern for demonstration

In practice, the selected floor pattern was converted to a video file and it was provided to the dancers and a trainer as a reference for their practice. The trainer who managed the practice gave us useful comments. The visualized floor pattern helped them remember their paths and corresponding motions and understand the big picture of the whole dance rather than the individual positions of dancers. However, as the choreographer interpreted and modified the floor pattern, the choreography was not exactly the same as the video, which confused the dancers a little. Usability will improve if a choreographer can manually modify a floor pattern shape just as he/she interpreted.

The piece was performed on stage with an audience (Fig. 17). Feedback was obtained from the audience after the performance. They were interested in many choreographic events happen frequently with complex combinations. This is a big characteristic of contemporary dance works compared to ballet. In order to create such a complex sequence in the production, it is common to let dancers improvise with fragmentary motivations and refine the outcome of random behaviors. It is an iterative process that takes a lot of time and energy. It is promising that we have been able to produce similar results in a short time using our system.

One of the audience in the front row said that he/she found it difficult to determine where to focus while another at the back row stated that he/she could appreciate the transitions on the entire stage. It suggests the importance of focus of a floor pattern, i.e., where a choreographer wants the audience to see the dance. It is a rather higher-level feature that include cognitive and theatrical aspect. With our current work as basis, it would be our future work to expand to such higher-level features.

Fig. 17 Demonstration performance scenes. Dancers are performing as one group (top left) or several parts (top right and bottom left). Collision of path induces interaction (bottom right)

4.1 Large-scale environment

Our system was mostly implemented and tested for situation with 3 to 6 dancers, which proper minimum number for showing collective and composite floor pattern. To see another potential of our system, we tested it with 18 dancers, which 3 times more. Firstly, our iteration system showed that its features converge successfully in the large-scale situation and one case was plotted in Fig. 18.

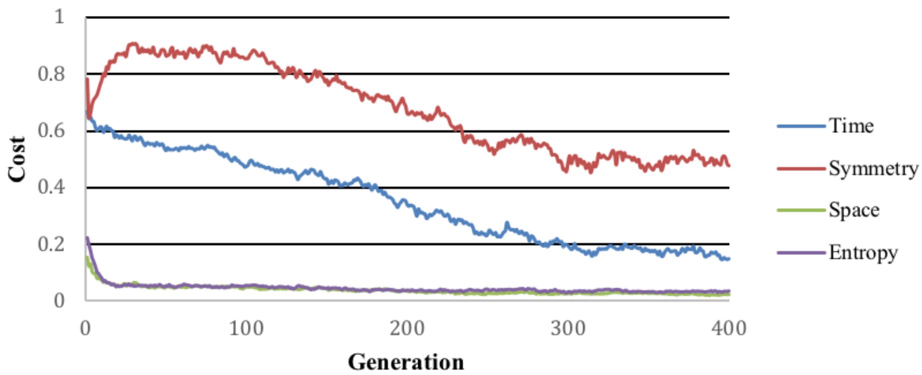


Fig. 18 An example of the time, space, symmetry, entropy costs change in evolution process for 18 dancers

We ran multiple iterations with different feature configurations and Fig. 19 shows some examples. From top to the bottom, each pair of floor patterns is result of slow and fast, symmetrical and asymmetrical, curved and straight, and ordered and chaotic features respectively. While it showed meaningful distinctive differences, it was thought too complicated and not enough geometric correlations between dancers, which are what people often expect from large-scale group dances. Applying mesh or hierarchical structure might improve such aspect in the future.

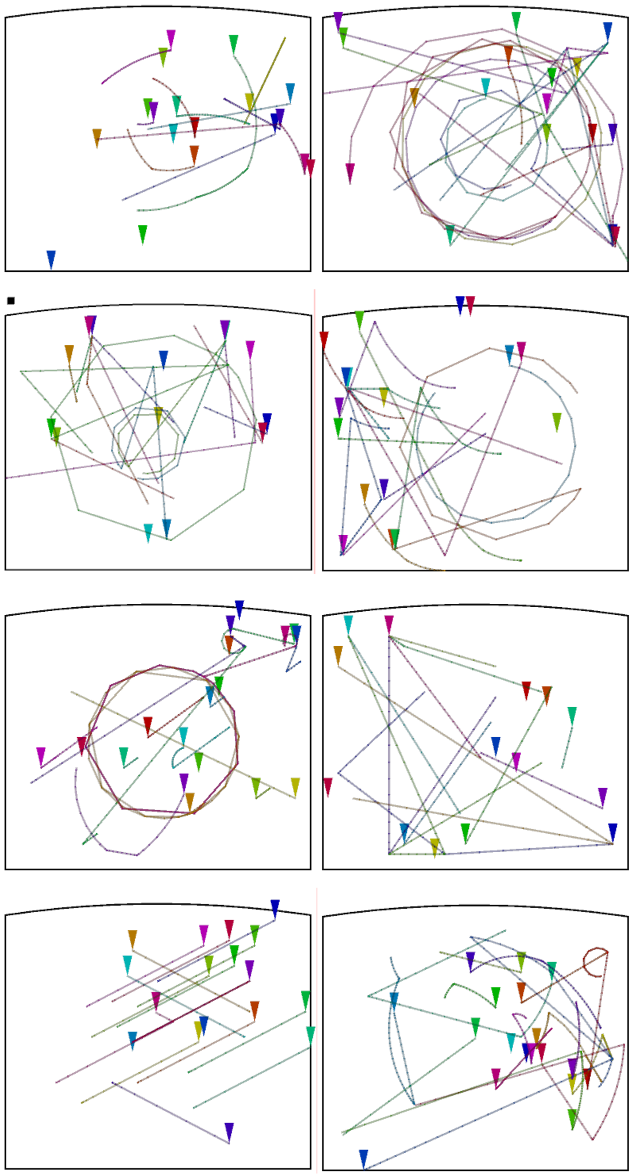


Fig. 19 Samples of generated floor patterns with 18 dancers

5 Discussion and Future Work

Aiming to assist choreographers, we developed a system that generates floor patterns for dance. Through a literature survey, we selected four major feature elements and modeled them mathematically. We defined our own model that efficiently expresses dance floor patterns. To successfully obtain diverse solutions that satisfy a choreographer's desires, we developed appropriate evolution schemes for the floor pattern.

User test shows that our system is capable of generating floor patterns that subjects recognize the intended differences of feature values. Our system provides choreographers with a set of floor pattern ideas that reflect their preferences, thereby improving the efficiency of the creative process of choreography. In addition, this system will help them explore outside their comfort zone and guide them to a novel creation methodology. While we communicated with dancers and reflected their opinions throughout our research process, our system has not been used in an actual dance production yet, which remains our next task.

Our work has a number of limitations, which can be interesting future research topics. First, the four features that we implemented can have more complex dimensions. For example, the desired value of time (speed) element may change with time. Space element may change with the gaze or gestures of dancers. In addition to the global symmetry that we implemented, partial symmetry, repetition [27], and temporal symmetry [6] might induce more geometric structures. In addition, while our system treated all features with equal importance, a choreographer might want to treat some features more importantly than others. Such preference control is another promising future work and Fang's work [36] can be referred to for this purpose. Also, as aforementioned, choreography is a composite process that choreographers should consider vast elements. For these kind of problems, machine learning-based approach would compensate the limitation of our rule-based approach. It will require acquisition of digitized path data of current dance pieces and extracting behavior information from the raw motion data like Liu's work [37].

Choreography is a complicated integrative process that requires consideration of numerous aspects. For instance, musical mood, flow, lyrics, and theatrical plot are important parts of the choreography, but we did not consider them in our work. The relation between such higher-level features and an actual floor pattern is not straightforward, and between them a large cognitive distance exists. One promising way to relate these higher-level features and a floor pattern would be to add an intermediate layer between them. Laban's theory is thought to be a proper tool for this because it was invented for scientific expression of dance. From this perspective, our work may serve as the mid-layer that abstracts high-level features from physical floor pattern data. Connecting our features to the higher-level features might be the next step of big challenge, which will require affective computing, semantic analysis, as well as diverse artistic studies. Before such an ideal choreographic tool becomes available, for many dancers with understanding of Laban's theory, we believe that our system provides intuitive parameters to achieve a certain floor pattern style that a choreographer wants.

References

1. Carlson K, Schiphorst T, Pasquier P (2011) Scuddle: Generating Movement Catalysts for Computer-Aided Choreography. Proc Second Int Conf Comput Creat
2. LaViers A, Chen Y, Belta C, Egerstedt M (2011) Automatic Sequencing of Ballet Poses. IEEE Robot Autom Mag 18:87–95 . doi: 10.1109/MRA.2011.942118
3. Gentry S, Wall S, Oakley I, et al (2003) Got rhythm ? Haptic-only lead and follow dancing. In: In Proceedings of Eurohaptics. pp 330--342
4. Lapointe F-J, Époque M (2005) The dancing genome project. In: Proceedings of the 13th annual ACM international conference on Multimedia - MULTIMEDIA ’05. ACM Press, New York, New York, USA, p 555
5. Laban R, Lawrence FC (1947) Effort. Macdonald & Evans
6. Lynne A, Tarin C (1982) The Intimate Act of Choreography. University of Pittsburgh Press, USA
7. Cunningham M, Jacqueline L (1985) The dancer and the dance: Merce Cunningham in conversation with Jacqueline Lesschaeve. Marion Boyars
8. Copeland R (2004) Merce Cunningham : the modernizing of modern dance. Routledge
9. Gentry S, Murray-Smith R (2003) Haptic dancing: human performance at haptic decoding with a vocabulary. In: SMC’03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme - System Security and Assurance (Cat. No.03CH37483). IEEE, pp 3432–3437
10. LaViers A, Egerstedt M, Chen Y, Belta C (2011) Automatic Generation of Balletic Motions. In: 2011 IEEE/ACM Second International Conference on Cyber-Physical Systems. IEEE, pp 13–21
11. Lee M, Lee K, Park J (2012) Music similarity-based approach to generating dance motion sequence. Multimed Tools Appl 62:895–912 . doi: 10.1007/s11042-012-1288-5
12. LaViers A, Egerstedt M (2012) Style based robotic motion. In: 2012 American Control Conference (ACC). IEEE, pp 4327–4332
13. Aristidou A, Stavrakis E, Charalambous P, et al (2015) Folk Dance Evaluation Using Laban Movement Analysis. J Comput Cult Herit 8:1–19 . doi: 10.1145/2755566
14. Loke L, Larssen AT, Robertson T (2005) Labanotation for design of movement-based interaction. In: Second Australasian Conference on Interactive Entertainment. Creativity & Cognition Studios Press, pp 113–120
15. Moon B (2008) Easily learning genetic algorithms. Hanbit Media, Seoul
16. Antunes R, Leymarie F (2012) Generative Choreography: Animating in Real-Time Dancing Avatars. In: Machado P, Romero J, Carballal A (eds) Evolutionary and Biologically Inspired Music, Sound, Art and Design SE - 1. Springer Berlin Heidelberg, pp 1–10

17. Vircikova M, Sincak P (2010) Dance Choreography Design of Humanoid Robots using Interactive Evolutionary Computation. 3rd Work Young Res Human-Friendly Robot 2010)
18. Eisenmann J, Lewis M, Parent R (2016) Spatiotemporal Ideation & Generation with Interactive Evolutionary Design. Leonardo 49:246–250 . doi: 10.1162/LEON_a_01102
19. den Heijer E, Eiben AE (2012) Maintaining Population Diversity in Evolutionary Art. In: Evolutionary and Biologically Inspired Music, Sound, Art and Design. Springer, Berlin, Heidelberg, pp 60–71
20. Goldberg DE (1989) Genetic algorithms in search, optimization, and machine learning. Addison-Wesley Longman Publishing Co., Inc.
21. Kwon T, Lee KH, Lee J, Takahashi S (2008) Group motion editing. ACM Trans Graph 27:80 . doi: 10.1145/1360612.1360679
22. Henry J, Shum HPH, Komura T (2012) Environment-aware real-time crowd control. 193–200 . doi: 10.2312/SCA/SCA12/193-200
23. Johansson A, Helbing D (2007) Pedestrian flow optimization with a genetic algorithm based on Boolean grids. In: Pedestrian and Evacuation Dynamics 2005. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 267–272
24. Humphrey D, Pollack B (1987) The art of making dance. Princeton Book Co.
25. Woo KH (2000) When dance and music meet, 1st ed. Yesol, Seoul
26. Guest AH (2005) Labanotation: The System of Analyzing and Recording Movement. Psychology Press
27. Mitra NJ, Guibas LJ, Pauly M, et al (2006) Partial and approximate symmetry detection for 3D geometry. ACM Trans Graph - Proc ACM SIGGRAPH 2006 25:560 . doi: 10.1145/1179352.1141924
28. Besl PJ, McKay ND (1992) A Method for registration of 3-D shapes. In: Schenker PS (ed) Proceedings Volume 1611, Sensor Fusion IV: Control Paradigms and Data Structures. International Society for Optics and Photonics, pp 586–606
29. Rusinkiewicz S, Levoy M (2001) Efficient variants of the ICP algorithm. In: Proceedings Third International Conference on 3-D Digital Imaging and Modeling. IEEE Comput. Soc, pp 145–152
30. Mitra NJ, Gelfand N, Pottmann H, Guibas L (2004) Registration of point cloud data from a geometric optimization perspective. In: Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing - SGP '04. ACM Press, New York, New York, USA, p 22
31. Pires E, Machado J, de Moura Oliveira P (2013) Entropy Diversity in Multi-Objective Particle Swarm Optimization. Entropy 15:5475–5491 . doi: 10.3390/e15125475
32. Oulasvirta A, Roos T, Modig A, Leppänen L (2013) Information capacity of full-body movements. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13. ACM Press, New York, New York, USA, p 1289
33. Sareni B, Krahenbuhl L (1998) Fitness sharing and niching methods revisited. IEEE Trans Evol Comput 2:97–106 . doi: 10.1109/4235.735432

34. Cohoon JP, Paris WD (1987) Genetic Placement. IEEE Trans Comput Des Integr Circuits Syst 6:956–964 . doi: 10.1109/TCAD.1987.1270337
35. Michalewicz Z (1996) Genetic Algorithms + Data Structures = Evolution Programs
36. Fang Z (2009) A Weight-Based Multiobjective Genetic Algorithm for Flowshop Scheduling. In: 2009 International Conference on Artificial Intelligence and Computational Intelligence. IEEE, pp 373–377
37. Liu Y, Nie L, Han L, et al (2015) Action2Activity: Recognizing Complex Activities from Sensor Data. In: Proceedings of the 24th International Conference on Artificial Intelligence. AAAI Press, pp 1617–1623