Visualizing Dance Formations:

The Choreographer's Tool

Marwa Nur Muhammad

An Undergraduate Senior Thesis

Adviser: Professor Dianna Xu

Department of Computer Science

May 2009

Contents

1 Introduction

Many dance forms like bhangra, classical Indian, hip-hop etc. that have multiple dancers on stage at any time rely heavily on the patterns or formations made by the way dancers position themselves. Choreographers often struggle to communicate the kind of patterns they want the dancers to form. While there is software to help visualize body movements, very few options are available that help visualize the position of all the dancers and the overall patterns formed in the dance.

The goal of this project is to develop software to help choreographers who are creating dance pieces for a team of multiple dancers to visualize stage formations. It also helps the dancers understand where their individual positions should be and what each formation is supposed to look like as a whole. Using a time line, the user or choreographer notes what the positions of the dancers will be at different points in the dance. The software then comes up with easy and fast transitions from one formation to the next, helping dancers visualize how and where they need to move in the dance. Additionally, based on the number of people on stage at a given point, the software suggests frequently used formations to help the choreographer come up with patterns.

2 Related Work

Similar work has been done to visualize dance formations in the past. Dance Formation Maker was done as an individual school project [1]. Dance Forms created by Credo Interactive is a commercial software and is based on Life Forms Dance software [2, 3].

2.1 Dance Formation Maker

Dance Formation Maker was a project that was created to provide an effective Flash-based tool for choreographers of formation-dependent dance routines [1]. It was created in April 2006 by Rahul Kak, a student at Duke University, as a project to facilitate choreography for an entourage dance team that he was part of. Figure 1 shows a screen shot of the Formation Maker interface.

Dance Formation Maker bears similarities to the project described in this paper. It gives a birds-eye view of the stage and each dancer is represented by a dot. The color of the dancer can be chosen by the user, and each dancer can also be tagged or given a short name for easy identification. A dancer is added by hitting an "Add dot" button. This can be really time consuming if the total number of dancers is large.

The program requires a music piece to be exported in XML format and a time line is

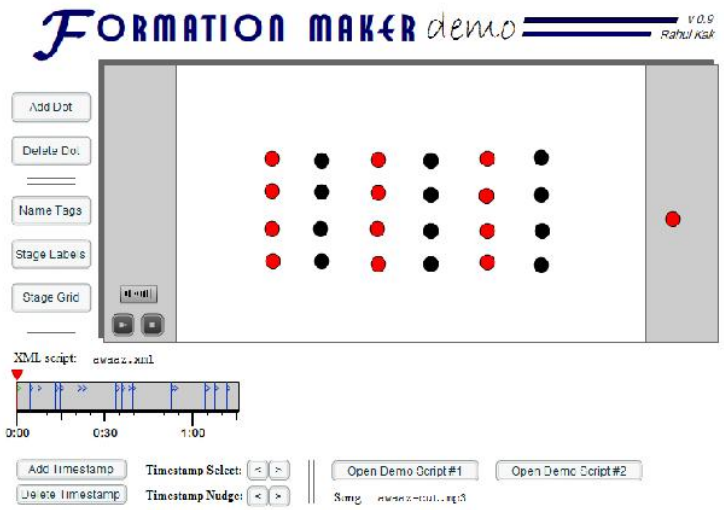

Figure 1: Formation Maker

created based on the length of the imported song. The user can "stamp" the time line at any point and save positions of the different dancers at those points. The software figures out what path each dancer has to take to move from one position to the next between consecutive stamps by calculating the shortest linear distance between them, without accounting for collision detection.

While the software is pretty simple and easy to use, it does not take into account a choreographer's point of view and find out aesthetically pleasing transitions between formations rather than the mathematically simple ones.

2.2 Credo Dance Forms and Life Forms Dance Studio

Dance Forms 1.0, created by Credo Interactive, is software that specializes in 3D character animation [2]. It is a software meant to help in complete choreography of modern and ballet dances primarily. It demonstrates entire dance scores with music integrated and is based on Life Forms, which is a motion capture and animation software [3]. While the main focus of both Dance Forms and Life Forms is in creating complete dances including the limb and torso movements of the dancers, it is possible to look at the spatial patterns created by using figures that move from one place to another but not animating their body movements.

Since both these software deal more with body movements than formation changes, they are far too complicated to use if the user wants to look at patterns formed by the positions of the dancers only. They are meant to be used particularly in modern and ballet dances

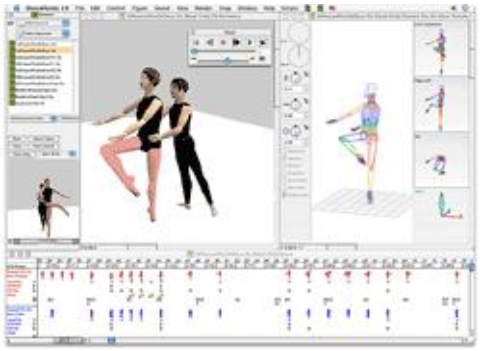

Figure 2: Dance Forms

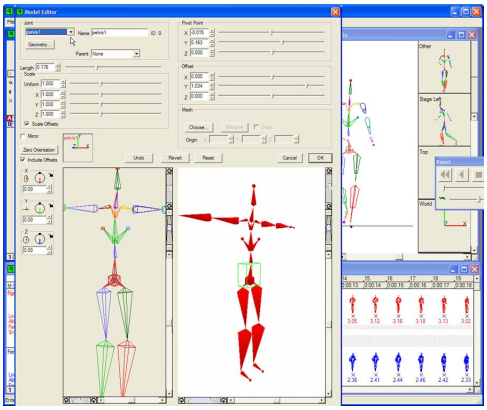

Figure 3: Life Forms

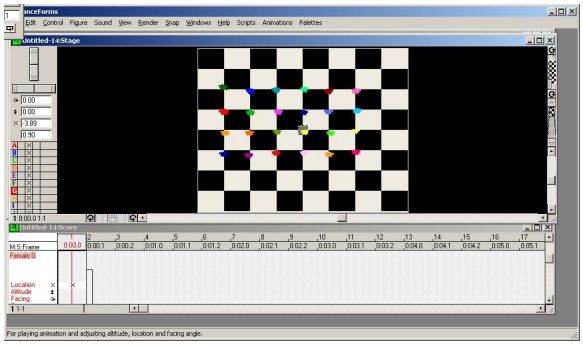

Figure 4: Top View of Dance Forms

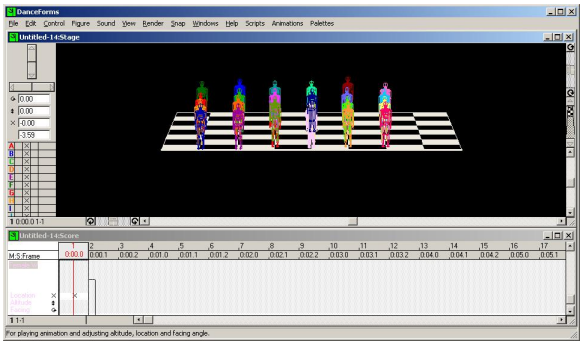

Figure 5: Front View of Dance Forms

in which formations are not a major component of the dance. Both Dance Forms and Life Forms generally deal with few dancers on stage and are difficult to use for multiple dancers. Thus, even though it is possible to use these software to visualize dance formations, they are not efficient for this purpose.

2.3 Visualizing Dance Formations

The goal of this project is to create software that will concentrate on making formations only and not the body movements of the dancers. The software is similar to Dance Formation Maker in many ways. It has a simpler interface with almost no learning curve and is easier to use for groups with large number of dancers. Each dancer is assigned a unique color, which the user can change if desired. The initial idea of the software interface is shown in Figure 6, an image designed by our friend also interested in such a software.

The next few sections describe the plan and tasks in detail, along with how the imple-

mentation was done.

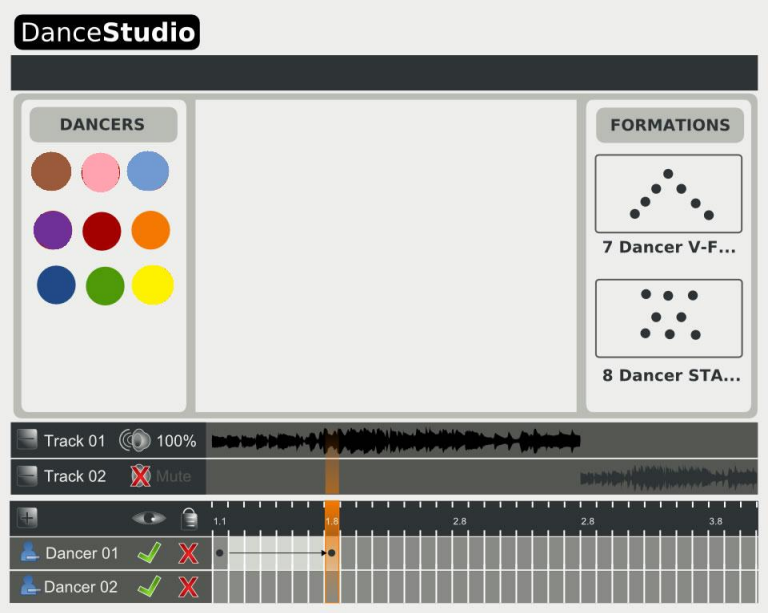

Figure 6: Dance Studio Interface Design, Image taken from [4]

3 Tasks and Analysis

The proposed software is called Dance Studio. Its four main components are as follows:

1. Basic interaction, which consists primarily of specifying the maximum number of dancers and being able to drag and drop any dancer.

2. Saving formations, which is done with the help of a time line.

3. Transition calculations, which are done by the software itself to figure out how dancers move or transition from one formation to the next.

4. Suggested formations, which the software comes up with based on the number of dancers on stage.

3.1 Basic Interaction

The user begins by entering the maximum number of dancers that are to be in a dance piece. The maximum allowed by the software is 40. Each dancer is represented by a small circle which has a unique color. The dancers are displayed on the left side of the screen which is off-stage space. Each dancer can be dragged and dropped onto the part of the screen that represents the stage. The user is only allowed to drag the dancers around within the stage boundaries.

3.2 Saving Formations

A time line is created to keep track of when each formation is supposed to be made. Since dances are more regularly choreographed based on beats (often referred to as *counts*) rather than just the music itself, the time line is based on counts. The user is able to enter the number of beats or counts in the dance so that the time line can be built based on that number. There is a default number of time bars that appears if it is not specified and the user is able to change this number at any point of the dance since it is difficult to know how many counts there are in the dance at the very beginning. When a user specifies a certain formation, he clicks on the time bar at which that formation is to be created and hits a "Save" button to store that formation for that particular time bar. He can do this to as many time bars as he wants.

3.3 Transition Calculations

The software looks at two consecutive formations and figures out how the dancers should transition from one formation to the next. It does this by calculating the paths to take for each dancer, linear or curved, avoiding collisions and spans the number of time bars that are between those two consecutive formations. The user also enters the tempo of the music so that the software knows how fast the movements and transitions need to be.

3.4 Suggested Formations

Based on the number of dancers on stage, the software also suggests frequently used symmetrical patterns that the choreographer could use for the piece. Some frequently used patterns in choreography include circles, diagonals, staggered lines, vertical or horizontal lines etc. Formations generally differ significantly based on whether the number of dancers on stage is an even or an odd number and the software takes that into account when suggesting patterns.

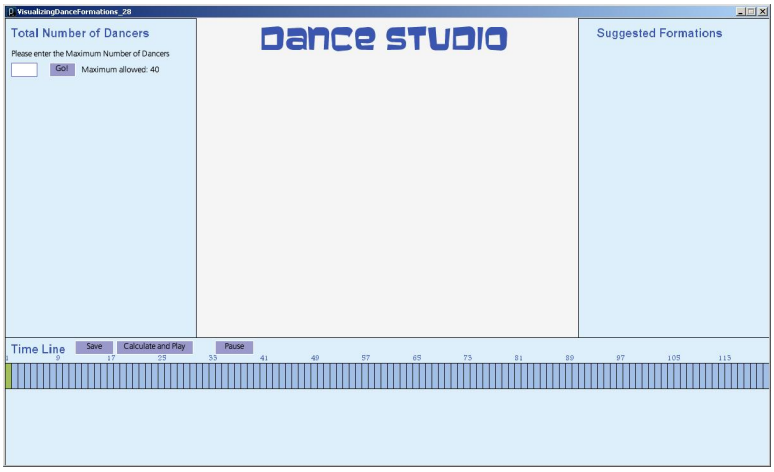

Figure 7: Dance Studio: Startup Screen

4 Implementation

The software that we developed is called "Dance Studio" and it is written in Processing [5], which is a language built on top of the Java programming language [6]. Processing is an environment that gives a visual context and is thus best used to program images, animation and interactions. Figure 7 and 8 show an overall picture of what the Dance Studio software looks like when it is first opened and when it is being run.

The implementation was done in steps. At first, we worked on implementing the interface with a stage in the center, a box on the left side that will show all the dancers in a dance, a suggested formations box on the right, and a time line at the bottom to keep track of formations corresponding to time frames in the dance. We worked on the software's functions after that, which includes saving formations and calculating transitions.

4.1 Basic Interaction

When the user first opens the program, a screen like the one in Figure 7 is shown. On the top-left corner, he will need to enter the maximum number of dancers in the dance, as shown in Figure 9. The number of dancers on stage at any particular time can vary, but the user will only have to enter the maximum number of dancers for that dance piece in the text box.

Once the user presses the "Go!" button, the selected number of dancers shows up in the left box, ready to be dragged and dropped onto the stage area. The program takes the user's input and calculates how the dancers should be spread out in a circular formation in the left box, as illustrated in Figure 10.

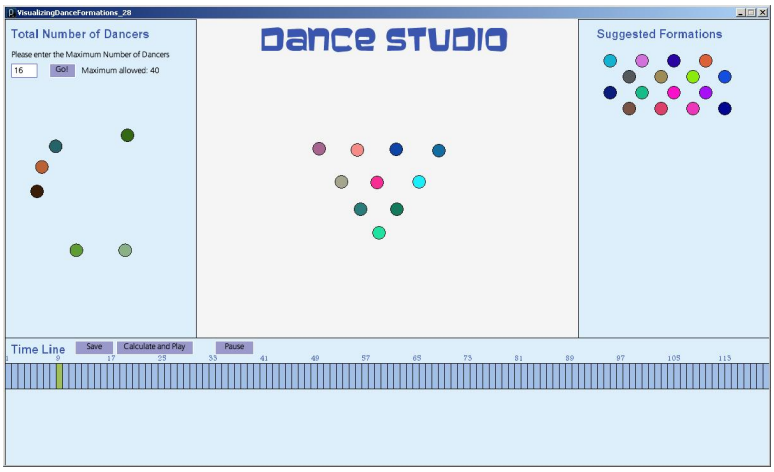

Figure 8: Dance Studio

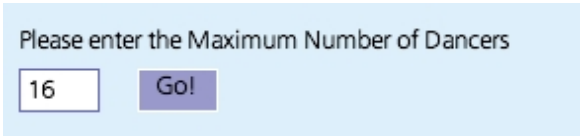

Figure 9: Entering Maximum Number of Dancers

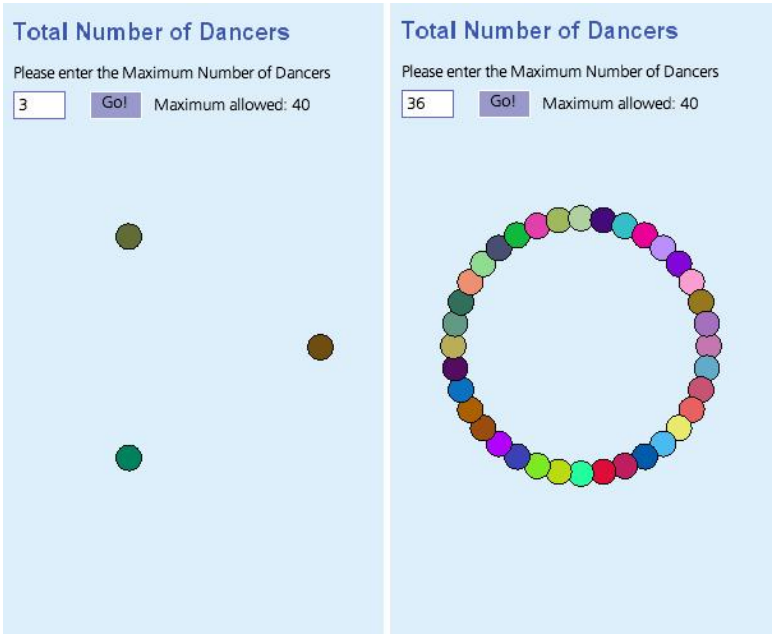

Figure 10: Left Box in Dance Studio

Figure 11: Time Line

4.2 Saving Formations

At the bottom of the screen is a time line containing small boxes representing each count or time frame, as shown in Figure 11. Users can drag and drop the dancers using the mouse onto the center area, which represents the stage, and save the formation for a particular time frame by clicking the "Save" button. When the button is clicked, the positions of each of the dancers are saved for the time frame that is selected. Each time frame holds x and y co-ordinates for each of the dancers. The user may save formations for as many time frames as he wants.

4.3 Transition Calculations

Once the user saves all the positions for all the desired time frames, he might want to visualize the transformations. He can do this by hitting the "Calculate and Play" button. When this button is clicked, the software calculates how the dancers should move to go from one formation to the next in a smooth and linear motion by taking the shortest linear path. These intermediate positions are saved into the intermediate time frames. The software then goes through each time frame and shows the user the positions for each dancer for that frame, whether the position was specified by the user or was calculated as part of a transition by the software. Figures 12 through 17 demonstrate the transition positions that were calculated by the software based on the formations specified for the first and sixth time frame: staggered lines for the first frame and an inverted V-shape for the sixth frame.

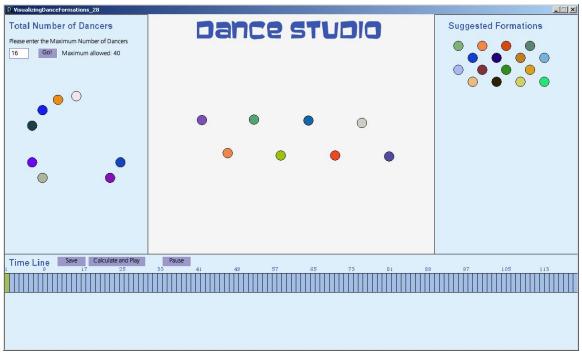

Figure 12: Frame 1

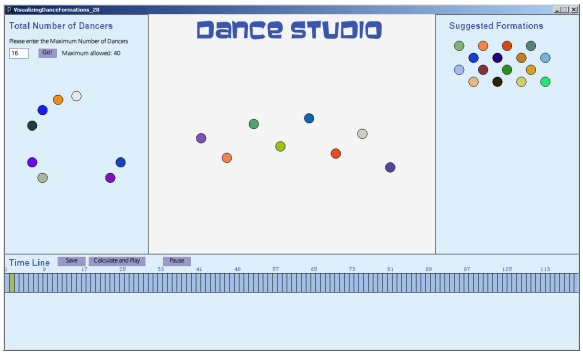

Figure 13: Frame 2

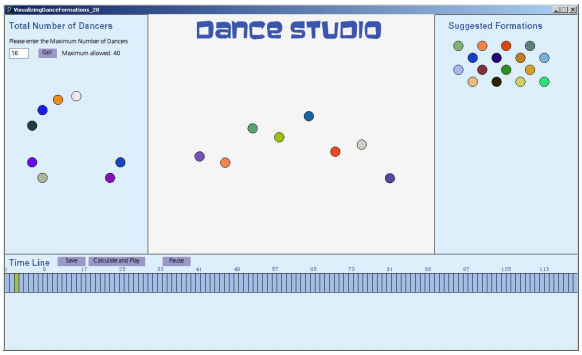

Figure 14: Frame 3

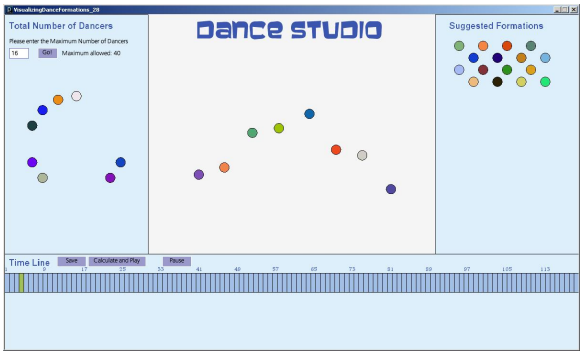

Figure 15: Frame 4

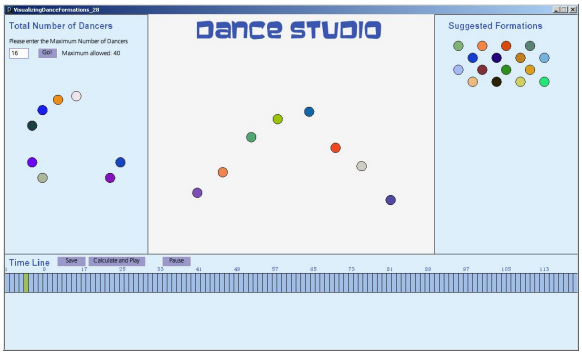

Figure 16: Frame 5

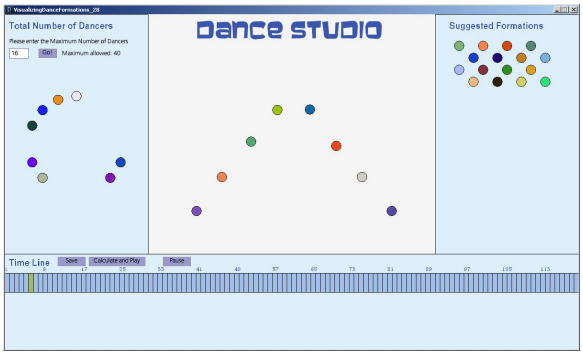

Figure 17: Frame 6

Choreographers might not like the way the transitions from one formation to the next is calculated by default in this software. This program only calculates the shortest linear path, but users might want a more aesthetically pleasing route. In that case, they are encouraged to save formations for some of the intermediate time frames to get their desired motions. Also, if the user does not like the automatic calculated positions, he can always overwrite it by specifying his own formations for those time frames. He can specify formations for all the time frames if he wants to, in which case the software will not have to calculate any transitions at all.

4.4 Suggested Formations

As the user enters the number of maximum dancers, the right hand box looks at symmetry and suggests a formation. It would have been ideal if the software came up with multiple suggested formations based on the number of people on stage, and the user could select a formation to have the dancers position themselves accordingly. However, in this version,

the implementation suggests only one formation based on the maximum number of dancers entered by the user. It is discussed in further detail in Section 5.4.

Code for this implementation is available in the Appendix section.

5 Results and Current Status

The current version of Dance Studio includes most of the basic functionalities from the original plan, although there are some limitations. The current status of the project is discussed in this section and future work is discussed in the next section.

5.1 Basic Interaction

When a user starts the program, he is able to enter the maximum number of dancers, and once the "Go!" button is hit, the dancers show up in the left box. The maximum number allowed by Dance Studio is 40 so that the screen does not appear too crowded. Once the user has entered a value once, he cannot change it to enter a second value; the program will exit in that case. The dancers are created with a random color assignment. In the current version, the user is not able to change any dancer properties including the color or visibility. Thus, if he is playing a sequence to see how the formations change and wants to look at the position changes of a few select dancers, there is not a way of making the other dancers invisible.

This version does not allow the user to enter any other information either, like the number of counts or time frames, or even the music tempo. The number of frames in the time line is set to the default value. Also, the music cannot be imported so the user has to choreograph the piece based entirely on counts.

5.2 Saving Formations

As was mentioned in section 4.2, the user can save formations for as many time frames as he wants. When the user saves a formation for a particular time frame, that formation is saved for all the time frames that follow the one for which it was saved, until a new formation is saved onto one of those time frames that come after that previously selected frame. If, however, the user creates a formation for a specific time frame but does not click "Save", that formation is saved for that particular time frame only and not for any of the frames that follow.

When the user saves a formation, he does not get a confirmation saying that the formation was successfully saved. At the same time, if a user forgets to save, no warning pop-up message appears to confirm if the user does not want to save the formation after all.

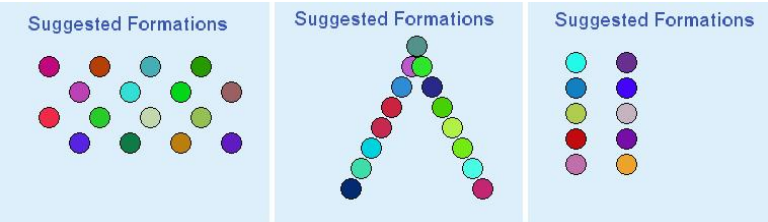

Figure 18: SuggestedFormations

5.3 Transition Calculations

The software calculates transitions based on the shortest distance each dancer has to travel from one formation to its consecutive one. Thus, it comes up with only one possible option for each transition. This does not give the user too many options. If they do not like the automatic transition calculation, they will have to specify formations for those intermediate frames themselves.

Also, there is no collision detection, so the software might suggest a path in which multiple dancers cross the same path at the same time. It does not give any warning even if there is a collision. Part of the goal of this project was to be able to suggest transitions that will never have any collisions but due to time constraints, this was not implemented. It is discussed more in the future work section.

5.4 Suggested Formations

Only one formation is currently suggested in the Suggested Formations box based on the maximum number of dancers in the dance and not on the number of people on stage at a point. The formation is based on whether the number is odd or even, and if even, whether it is divisible by 4. For an odd number of dancers, a V-shape formation is suggested. For an even number of dancers that is divisible by 4, the software suggests staggered lines with 4 dancers in each line. For an even number of dancers that is not divisible by 4, the formation that is suggested is two straight lines with equal number of dancers in each line. These formations are shown in Figure 18. Currently, there is no variation in the suggested formation; it is always going to be one of these three basic shapes.

The user cannot click on the suggested formation. Originally, we had wanted these suggested formations to be clickable such that the dancers organize themselves based on the formation that is being chosen by the user. Once again, due to time constraints, this was not implemented.

6 Future Work

The current status of Dance Studio has opened a lot of new opportunities for future work. These include not only the portions that we originally planned but were not able to incorporate into this version, but also improvements that we thought about along the way. The future work is listed in two portions: Level 1, which are the very next steps, and Level 2, which are steps to be taken in the future to improve the software's functionality.

6.1 Level 1

6.1.1 Music tempo

The music tempo is set to its default value in the current version of the software. In the future, we hope to ask the user what the tempo of the music is, so that the software can determine how fast the dance transitions should be. This will make the choreography appear more true to the actual dance.

6.1.2 Number of frames

Currently, the number of time frames that show up are set strictly to the default value. The user is not able to change it. In the future, the number of time frames will be a value that the user will input, the same way they input the number of dancers. A scroll bar will appear if the time line is too large to hold horizontally on the screen. Users will be able to alter this input number at any point.

6.1.3 Error Messages

The maximum number of dancers that the user can enter is set to 40. A larger number will not be well accomodated on the screen. If the user enters a number larger than 40, everything on the screen is deleted and the user is allowed to enter a new number. No error messages appear, however. In the next version of the software, there will be an error message to make it clear to the user why their input number was not accepted.

6.1.4 Dancer properties

As was initially planned, each dancer will have his own set of properties, including color, visibility, tags, etc. The user will be able to change those properties. For example, if he wants to see the motion of one particular dancer only throughout the piece, he can do so by changing the visibility property for all the other dancers to false. He can set the color of the

dancers to the color of the costumes the dancers will wear in the actual dance to visualize the dance more accurately. The choreographers can also tag the dancers based on their initials, which will help them keep track of exactly which person is in which position.

6.1.5 Add or Delete Dancers

Once a user enters the number of dancers at the beginning, it can no longer be changed. It would be great to have an option of adding or deleting dancers at any point to allow more flexibility.

6.1.6 Grids

In the current version, the dancers can be placed anywhere on the stage. In the future, we hope to have an option where the user can choose to have grids on the stage. With grids, the dancers will "click" into place whenever they are dragged and released on stage.

6.1.7 Confirmation

Currently, when a user saves a formation, there is no confirmation that the formation was saved successfully. At the same time, if a user has set a formation for a certain time frame but forgets to hit the "Save" button after that, there is no warning message. In the next version, there will be confirmation if a formation is successfully saved. At the same time, if the user sets a formation for a certain time frame but does not save it, the software will ask, "Are you sure you do not want to save this formation?"

6.1.8 Off-stage space

There is a stage space in the center in this version of Dance Studio, but there is no off-stage space on either side. If users want their dancers to be off-stage on a certain side of the stage, they will have to be lined along the edge of the stage area. We hope to have "curtains" on either side that will represent off-stage area, the way it is in real-life stages.

6.1.9 Saving

Currently, the user can make a choreographed piece or file but there is no way of saving that file so that it can be opened or used in the future. This option will be made available in the next version of Dance Studio.

6.1.10 Suggested formations

We hope to have the software suggest more formations, including ones that are complex, based on the number of dancers on stage at any point. The user should be able to click on any of those suggested formations and the dancers will organize themselves into that formation on the stage area.

6.2 Level 2

6.2.1 Music Import

We hope to allow music to be imported into the software in some of the most frequently used formats. This will help the user know how long the time line should be and determine the music tempo. It will also be a lot easier to see the choreography when the music and movements are synchronized.

6.2.2 Transition Options

The software will come up with multiple choices for the transitions from one formation to the next. There will always be linear motions and curved motions, among others. The linear transition will be selected by default, but the user will have the option to look at and change the transition he wants for every formation change sequence. This will increase the aesthetics of the dance.

6.2.3 Collision Detection

In a future version of Dance Studio, collision detection between dancers will be enabled. If the users are specifying a formation that has two dancers in the same place at the same time, they will receive a warning. Also, if the software comes up with a transition that includes collisions, it will tweak the algorithm so that the collision is avoided.

7 Acknowledgments

I would like to thank the following people for their help and support: Professor Dianna Xu, Computer Science Department, Bryn Mawr College, for being my thesis adviser and helping me with the project all throughout; Professor Deepak Kumar, Computer Science Department, Bryn Mawr College, for being a second adviser and giving me feedback about my project every week; Madeline Cantor and Linda Haviland Caruso, Dance Department, Bryn Mawr College, for giving me the choreographer's perspective; Professor Victor Donnay, Mathematics Department, Bryn Mawr College, for giving me a mathematical point of view for calculating transitions; Professor Ira Greenberg, Interactive Media Studies Department, Miami University, for all his help with the Processing language; Developers of Processing for creating software that makes building graphical interfaces simpler; Julia Ferraioli, Software Engineer, Microsoft, for going over my thesis and giving me valuable feedback, and Sunthar Premakumar, School of Engineering, University of Pennsylvania for helping me come up with a concrete idea for my project.

References

[1] Dance Formation Maker, Rahul Kak, 2006, http://mimi.cs.duke.edu/ kak/fmaker/

[2] Dance forms, Credo Interactive Inc, 2005, http://www.charactermotion.com/danceforms/

[3] Life forms, Credo Interactive Inc, 2005, http://www.charactermotion.com/products/lifeforms

[4] Emailed by Sunthar Premakumar, School of Engineering, University of Pennsylvania, 2009

[5] Processing Software, Ben Fry and Casey Reas, 2001, http://processing.org/

[6] Java Programming Language, Sun Microsystems, 1995, http://www.java.com