Master of Science in Informatics at Grenoble
Master Informatique
Specialization Graphics, Vision, and Robotics

Sketch-Based Posing, Animation, and Interaction of Multiple Characters: Animating Dancing Couples

Sarah Anne Kushner

Wednesday June 21, 2017

Research project performed at Inria Grenoble – Rhône-Alpes

Under the supervision of:
Marie-Paule Cani and Rémi Ronfard

Defended before a jury composed of:
James Crowley
Jury member 1
Jury member 2

June 2017

Abstract

Your abstract goes here...

Résumé

Your abstract in French goes here...

Acknowledgement

I would like to officially say thank you to those without whom this thesis would not have been possible. It brings me joy to acknowledge the following people.

Thank you to David Breen, Marcello Balduccini and Daryl Falco from Drexel University for being not only excellent and inspiring mentors and teachers, but also for helping and supporting me on my journey to study in France.

James Crowley deserves credit for making my move to France viable, and putting up with my numerous emails. I would also like to thank him for teaching me in the first semester of M2 along with all the other MoSIG M2 professors.

I would like to express my sincere gratitude to both my supervisers Marie-Paule Cani and Rémi Ronfard for their invaluable assistance, guidance, and patience through the process of a Master thesis.

The engineers Maguelonne de Brive and Julien Daval

I have so much appreciation for Tom. Thank you for letting me think out loud to you on the phone; comforting me when I freak out to you on the phone; sending me pictures of our baby bunny, Cinnamon Louise Bun; and providing me with constant encouragement all the way from the United States.

It wouldn't be fair not to mention and thank my dear dancer friends Ruth and Vivian for answering my sometimes obvious and weirdly specific questions about your experiences in dance and choreography.

Lastly, thank you to the jury for reading this report and listening to my presentation.

Contents

— 1 —

Introduction

1.1 Background

3D animation can be a painstakingly tedious activity. To create a desired animation, animators go through the long process of keyframing. Keyframes are set positions that define the start and end points of a movement, sequences of poses which are transformed in time. Typically, animators assign poses to certain frames over time, so that in-between motions can be generated by a computer. To get an accurate animation, artists usually must assign many keyframes, then spend time adjusting and editing them to be more precise. The fact that industry professionals take so much time and effort to do this shows that for an amateur or untrained artist, creating *good* 3D animation is close to impossible.

In Figure 1.1, the highlighted object – the ball flying through the window – has keyframes attached to it on the timeline, which represent its various positions and rotations in time. Even in a simple bouncing ball animation like this one, relatively speaking, many keyframes are required: 20 keyframes for a simple object to move within seven seconds. I invite the reader to imagine the amount of keyframes an articulated character might require, considering each joint is its own object.

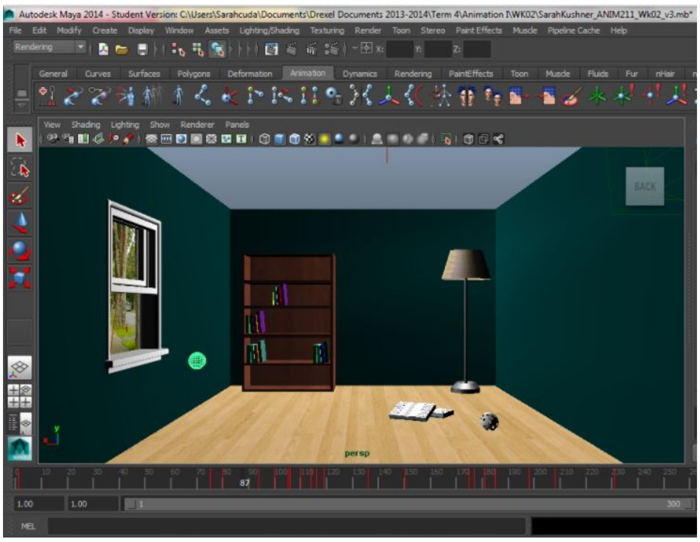
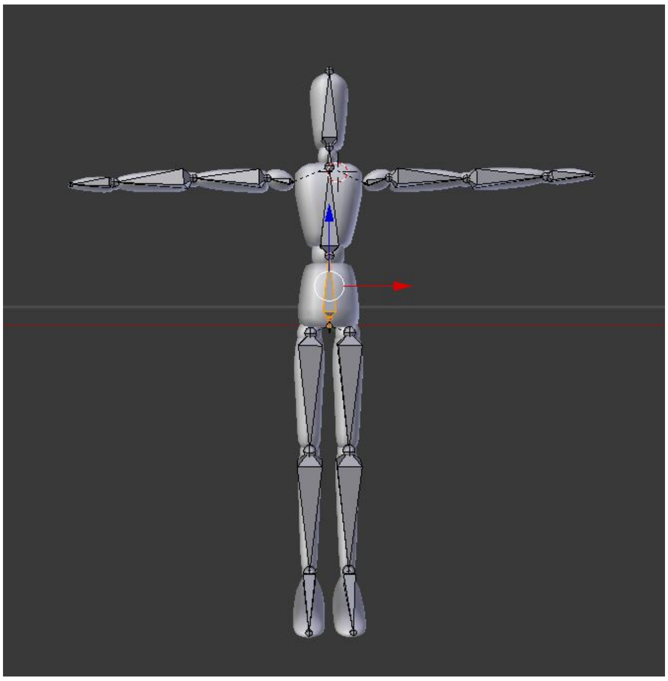


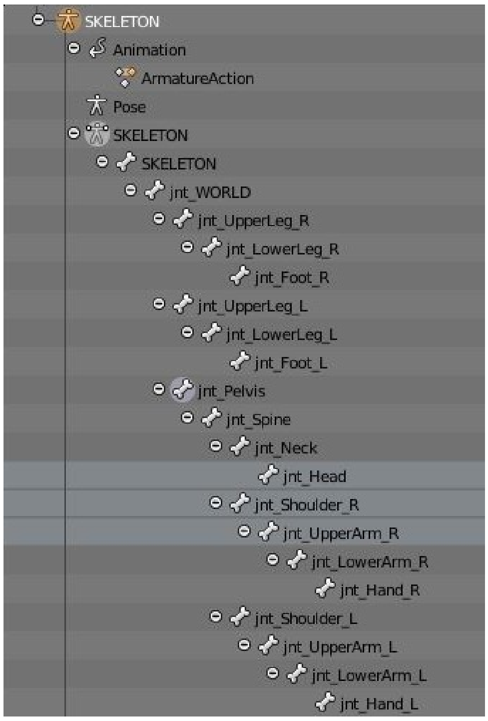
Figure 1.1 – Example of keyframing in Autodesk Maya. Each of the red lines on the timeline at the bottom indicate a keyframe for the highlighted object.

1.2 Problem Statement

Among the most complicated characters to animate in 3D animation are humanoid characters. To ease this task, animators create a skeleton for their character called a rig, that consists of joints connected by bones to give a structure to the character. Humanoid rigs can range in complexity from (somewhat) simple to extremely complicated depending on the amount of detail desired by the user. The structure is a hierarchy of joints that can also be seen as a tree with a root, which in the humanoid case, is usually the pelvis. The leaf nodes of this tree, which are loacted at the maximal parts of the body, are the hands, feet, and head of the character. Leaf nodes come at the end of a kinematic chain, which can be followed back up to the root.



(a) Example of a humanoid skeleton.



(b) The hierarchy of joints corresponding to the skeleton.

Figure 1.2 – Humanoid skeleton shown in Blender.

1.2.1 Kinematics

Forward and inverse kinematics are two general animation methods used mainly in situations in which articulated characters need to move according to some contraints. In order to animate this structure successfully, controls are added that allow for forward and inverse kinematics. These controls help the animator move the character into poses that will then act as keyframes.

Forward kinematics (FK) is a method of calculating the position and orientation of the end of a kinematic chain (i.e. a hand or foot) given the positions and angles of the joints higher up in the chain all the way to the root.

Inverse kinematics (IK) is the method opposite of forward kinematics. That is, the goal is to calculate the angles and positions of joints in the chain, given the angle and position of just the end of the chain. This goal is much harder to reach, seeing that more information needs to be calculated than is given. This problem is underconstrained. Many IK algorithms exist to calculate joint angles and positions, which will be discussed in chapter 3.

1.2.2 Multiple Characters

The animation of multiple characters, along with all the previously mentioned challenges, comes with its own unique set as well. The line of action technique works extremely well for a single humanoid character, and even multiple humanoid characters separate from each other. The problem is discovered when the humanoid characters interact, when they are in close proximity to each other or when they touch each other.

Occlusion and collisions can both become issues, especially when more than one character is involved.

1.2.3 Posing

posing

1.3 Scientific Approach and Investigative Method and Results

The use case for this research in multi-character animation is a dancing couple. There are many combinations of poses a human can be in, let alone two humans *and* the two humans interacting as one.

I traced dancers. traced keyframes. drew lines of action all over videos.

Found patterns.

Found common poses.

developed

1.4 Contents of this report

In the following chapter (chapter 2), I cover the state-of-the-art for this particular problem. I discuss a brief history of dance notation – how choreographers and dancers use sketching on paper to brainstorm and communicate their ideas of motion, formation, and pose of dance. Then I will talk about the existing sketch-based systems used for posing articulated characters, covering the benefits and limitations of each. A sizeable portion of my work has had to do with kinematic trees and graph data structures, so I will go into a few important graph theory algorithms. Generating animation from existing data is also a widely used technique relevant to my work.

In chapter 3, ...

— 2 —
State-of-the-Art

2.1 Dance

Sutton Dance Writing
Labanotation
Benesh Movement Notation

2.2 Sketch-Based Systems

The IMAGINE group at Inria has made it their mission to tackle this problem. They have made significant progress on a project where they aim to offer more intuitive tools to author 3D digital content. The IMAGINE team has invented (1) a type of notation made especially for posing and animating 3D characters (2) a technique for posing called the line of action, in which a user can draw a line in the shape they want a kinematic chain to take and (3) a technique for animation called space-time sketching, in which a user can draw a line in the path they want a model to take and it will be animated accordingly. As the character follows the path, its model bends and changes shape in a physically realistic way. Their system currently supports creating different movements with the path such as bouncing, rolling, and twisting.

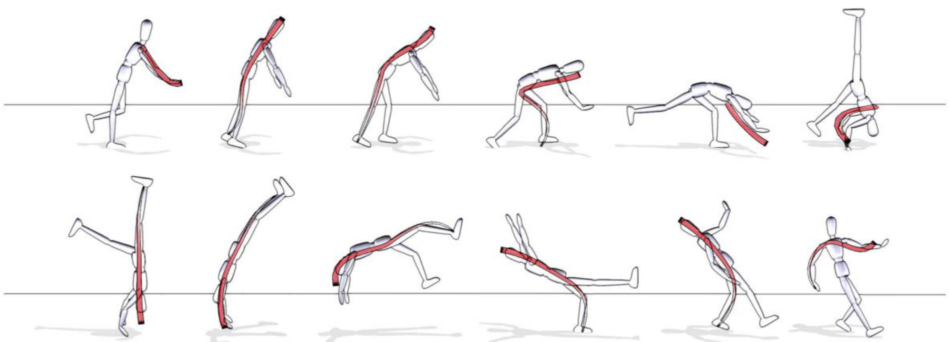


Figure 2.1 – One character's keyframes using the line of action technique.

The Line of Action: an Intuitive Interface for Expressive Character Posing – 2013
Adding dynamics to sketch-based character animations – 2015
Space-time sketching of character animation – 2015

Artist-oriented 3D character posing from 2D strokes – 2016
people in Switzerland also did the posing
Sketch to pose in Pixar's presto animation system – 2015

2.3 Graph Theory

Finding All the Elementary Cycles in a Directed Graph
A New Search Algorithm for Finding the Simple Cycles of a Finite Directed Graph
An Algorithm for Combining Graphs Based on Shared Knowledge

2.4 Generating Animation

FootSee: an Interactive Animation System
Footskate Cleanup for Motion Capture Editing
SketchiMo: Sketch-based Motion Editing for Articulated Characters – 2016
sketching for editing trajectories and poses
Retargetting Motion to New Characters
Using an Intermediate Skeleton and Inverse Kinematics for Motion Retargeting
Motion Graphs
Style-Based Inverse Kinematics – 2004
generative models for motion capture sequences used to build animations

Displacement constraints for interactive modeling and animation of articulated structures – 1994
fitting geometric constraints using physics

A constrained inverse kinematics technique for real-time motion capture animation – 1999
Dancing-to-Music Character Animation
Synthesizing Dance Performance Using Musical and Motion Features – 2006
between music and an animation generated from a motion graph built from motion capture

— 3 —

Theoretical Foundations of Animation

3.1 Skeletons, Rigs, and Controllers

3.2 In-betweening

3.3 The Line of Action

3.3.1 Notation

Taking inspiration from Sutton Notation, we propose a new notation for representing the poses of two characters both together and separately.

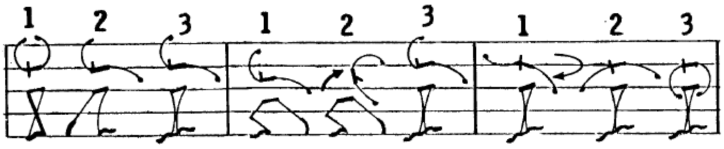


Figure 3.1 – An example of Valerie Sutton's Dancewriting.

3.3.1.1 Common Multi-character Poses

Figure 3.2 – Dancing in the Dark.

There is a dance scene in the film "The Band Wagon," where Cyd Charisse and Fred Astaire start out by walking together side by side, exchanging twirls until it morphs completely into a swing style dance. This scene is our use case for inventing a notation which extends seamlessly to more than one character.

To determine which poses for these dancers were common, I annotated the video with what I thought good keyframes would be if the two characters were treated as one.

There are 6 poses combinations for the characters separately. The asymmetrical poses for the odd number of LOAs can be mirrored (for 3 and 5 LOAs); therefore there are actually

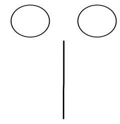
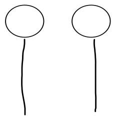
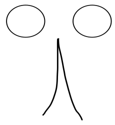
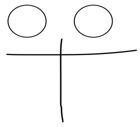
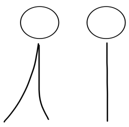
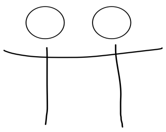
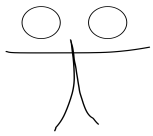
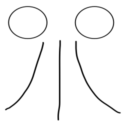
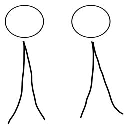
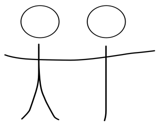
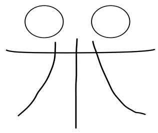
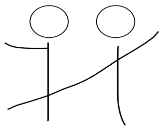
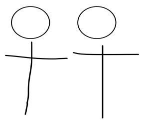
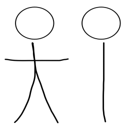
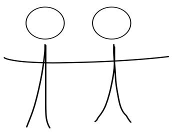
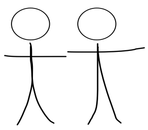
	Separate	Together		
1 LOA				
2 LOA				
3 LOA				
4 LOA				
				
5 LOA				
6 LOA				

Table 3.1 – Types of Keyframes for Two Dancing Characters

8 poses for two characters separately. There are 10 combinations for the characters together, treated as one character. There are 2 poses which can be mirrored, so there are actually 12 poses together. Overall, 20 poses exist for 2 articulated humanoid characters.

3.4 Space-time Sketching

3.5 Optimization Problems

The Conjugate Residual Method for Constrained Minimization Problems – 2015
Constrained Closed Loop Inverse Kinematics – 2010

3.6 Rig Combinations as Trees

— 4 —

Implementing a Character Connection Algorithm

4.1 Interface

4.2 Intelligently Connecting Rigs

Rather than changing the whole LOA concept and optimization algorithm, we reduce the more complex problem of using LOAs on multiple characters to the original LOA problem and utilize (nearly) the exact same minimization.

4.3 Challenges

4.4 User Constraints

Only allowed to connect and disconnect skeletons at keyframes.

— 5 —
Experimental Validation of Solution

5.1 Carefully Selecting Sample Keyframes

5.2 Establishing a Baseline for Comparison

5.3 Our Solution

5.3.1 Experiment

Describe the performance metrics, experimental hypotheses, experimental conditions, test data, and expected results. Provide the test data. Interpret the results of the experiments. Pay special attention to cases where the experiments give no information or did not come out as expected. Draw lessons and conclusions from the experiments. Explain how additional experiments could validate or confirm results.

5.3.2 Results

5.3.3 Conclusions and Future Experiments

— 6 —
Discussion

Discussion lessons learned from the experiments, and new problems that are raised.

— 7 —
Conclusion

Give a summary of the problem, approach, implementation and evaluation. Discuss the principal results in abstract terms. Discuss expected impact and further research directions. Explain how the project satisfies the evaluation criteria for a Masters Research project.

— A —
Appendix

A.1 Glossary

articulated character
definition

CCD
cyclic coordinate descent

constraint
cyclic coordinate descent

controller
cyclic coordinate descent

keyframe
cyclic coordinate descent

Bibliography

[1] Hisham Al-Mubaid and Said Bettayeb. An algorithm for combining graphs based on shared knowledge. In *Proceedings of BICOB 2012 International Conference on Bioinformatics and Computational Biology*, pages 137–142, 2012.

[2] Autodesk. Maya. `https://www.autodesk.fr/products/maya/overview`, 2017.

[3] Marguerite Causley. *An Introduction to Benesh: Movement Notation*. Ayer Company Pub, 1980.

[4] Byungkuk Choi, JP Lewis, Yeongho Seol, Seokpyo Hong, Haegwang Eom, Sunjin Jung, Junyong Noh, et al. Sketchimo: sketch-based motion editing for articulated characters. *ACM Transactions on Graphics (TOG)*, 35(4):146, 2016.

[5] Behzad Dariush, Youding Zhu, Arjun Arumbakkam, and Kikuo Fujimura. Constrained closed loop inverse kinematics. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 2499–2506. IEEE, 2010.

[6] Blender Foundation. Blender. `https://www.blender.org/`, 2006–2017.

[7] Jean-Dominique Gascuel and Marie-Paule Gascuel. Displacement constraints for interactive modeling and animation of articulated structures. *The Visual Computer*, 10(4):191–204, 1994.

[8] Michael Gleicher. Retargetting motion to new characters. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 33–42. ACM, 1998.

[9] Keith Grochow, Steven L Martin, Aaron Hertzmann, and Zoran Popović. Style-based inverse kinematics. In *ACM transactions on graphics (TOG)*, volume 23, pages 522–531. ACM, 2004.

[10] Martin Guay, Marie-Paule Cani, and Rémi Ronfard. The line of action: an intuitive interface for expressive character posing. *ACM Transactions on Graphics (TOG)*, 32(6):205, 2013.

[11] Martin Guay, Rémi Ronfard, Michael Gleicher, and Marie-Paule Cani. Adding dynamics to sketch-based character animations. In *Proceedings of the workshop on Sketch-Based Interfaces and Modeling*, pages 27–34. Eurographics Association, 2015.

[12] Martin Guay, Rémi Ronfard, Michael Gleicher, and Marie-Paule Cani. Space-time sketching of character animation. *ACM Transactions on Graphics (TOG)*, 34(4):118, 2015.

[13] Chris Hecker, Bernd Raabe, Ryan W Enslow, John DeWeese, Jordan Maynard, and Kees van Prooijen. Real-time motion retargeting to highly varied user-created morphologies. *ACM Transactions on Graphics (TOG)*, 27(3):27, 2008.

[14] Donald B Johnson. Finding all the elementary circuits of a directed graph. *SIAM Journal on Computing*, 4(1):77–84, 1975.

[15] Lucas Kovar, Michael Gleicher, and Frédéric Pighin. Motion graphs. In *ACM transactions on graphics (TOG)*, volume 21, pages 473–482. ACM, 2002.

[16] Lucas Kovar, John Schreiner, and Michael Gleicher. Footskate cleanup for motion capture editing. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 97–104. ACM, 2002.

[17] David G Luenberger. The conjugate residual method for constrained minimization problems. *SIAM Journal on Numerical Analysis*, 7(3):390–398, 1970.

[18] Mentar Mahmudi, Pawan Harish, Benoît Le Callennec, and Ronan Boulic. Artist-oriented 3d character posing from 2d strokes. *Computers & Graphics*, 57:81–91, 2016.

[19] Vincente Minnelli. *The Band Wagon*. Metro-Goldwyn-Mayer (MGM), 1953.

[20] Jean-Sébastien Monzani, Paolo Baerlocher, Ronan Boulic, and Daniel Thalmann. Using an intermediate skeleton and inverse kinematics for motion retargeting. In *Computer Graphics Forum*, volume 19, pages 11–19. Wiley Online Library, 2000.

[21] Toru Nakata, Taketoshi Mori, and Tomomasa Sato. Analysis of impression of robot bodily expression. *Journal of Robotics and Mechatronics*, 14(1):27–36, 2002.

[22] Takaaki Shiratori, Atsushi Nakazawa, and Katsushi Ikeuchi. Dancing-to-music character animation. In *Computer Graphics Forum*, volume 25, pages 449–458. Wiley Online Library, 2006.

[23] Takaaki Shiratori, Atsushi Nakazawa, and Katsushi Ikeuchi. Synthesizing dance performance using musical and motion features. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 3654–3659. IEEE, 2006.

[24] Ryan Stelzleni, Bret Parker, Tom Hahn, Sarah Shen, Dan McGarry, and Chen Shen. Sketch to pose in pixar's presto animation system. In *ACM SIGGRAPH 2015 Talks*, page 26. ACM, 2015.

[25] Valerie Sutton. *Sutton Movement Shorthand: Writing Tool for the Dance*. Movement Shorthand Society Press, 1979.

[26] Wen Tang, Marc Cavazza, Dale Mountain, and Rae Earnshaw. A constrained inverse kinematics technique for real-time motion capture animation. *The Visual Computer*, 15(7):413–425, 1999.

[27] L-CT Wang and Chih-Cheng Chen. A combined optimization method for solving the inverse kinematics problems of mechanical manipulators. *IEEE Transactions on Robotics and Automation*, 7(4):489–499, 1991.

[28] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016.

[29] Herbert Weinblatt. A new search algorithm for finding the simple cycles of a finite directed graph. *Journal of the ACM (JACM)*, 19(1):43–56, 1972.

[30] KangKang Yin and Dinesh K Pai. Footsee: an interactive animation system. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 329–338. Eurographics Association, 2003.