Motographicon: a system for computer choreography

Peter Rajka
rajka@nada.kth.se

Tanus Ungvary
ungvary@kacor.kth.se

Royal Institute of Technology, Dept. of Speech Communication and Music Acoustics, KACOR (Kineto-Auditory Communication Research group)

Abstract

This demonstration paper describes the most essential features of MOTOGRAPHICON — a system for computer choreography — in relation to the development of a structure-oriented method for choreographic composition. A notation system is a fundamental prerequisite for any kind of structural approach and analytical conceptualization with some degree of generality. While the question, performance and perceiving of movement structures is a hierarchical process, different notational systems are necessary for different levels of movement description and manipulation [Mito-Yam, 90]. The different notational systems, tools and applications in Motographicon are intended to respond to these demands.

1 Notation of Human Motion in Motographicon

Motographicon [Rajka et al., 92] includes several movement notation systems, supported by different applications: Script Notation (MOVE data format), Function Notation and the Symbolic Notation.

The Script Notation (MOVE) is the common language for data exchange between different notations and presentation systems within Motographicon.

The Function Notation (graphs) provides a clear parametric representation of movements on a basis level, suitable for manipulation and analysis of the separate parameters of each body part.

The advantages of the Symbolic Notation are the possibilities to read and understand movements, combined with structural orientation for the building of higher level analytic concepts.

1.1 The Script Notation (MOVE)

The MOVE data [Lundin, 92] is in pure text format with commands and arguments grouped together in manner similar to LISP. This data format is supported by a library of functions for reading, interpreting and writing such files. A basic movement command corresponds to a body part and the joint supporting the part. The command specifies the target position for a movement starting at the present time and a duration for that movement. This is synchronized by the present time timing commands. A sequence of commands can be grouped to define a named procedure. The procedure can then be used as a basic command with the difference that the parameters do not represent angles but durations and in some cases a modifier ("mirror").

The duration for procedure calls are given as a percentage of the restated durations in the procedure. The mirror parameter gives the left-right reversal of all movements in the procedure.

The procedure definition capacity in MOVE makes it possible to build movement libraries. The libraries can contain frequently used positions, for instance the normal ballet positions, but also longer movement scripts such as jumps and walks, etc. For composition and notation of modern dance, a library of movement themes and shapes specific to this dance form can be built and then used to test different ways of combining these elements in space and time. A dance rotated with procedures can be modified into a different style by using a different library of procedures.

1.2 Function Notation

The trajectories of a movement occur between consecutive body positions placed in time, as defined in a MOVE file, generated by the Symbolic Notation Editor (MoGoLym) or written in a text editor. Since the body positions are described by angles, a sequence of movement is represented by a sequence of consecutive trajectories, i.e. by a set of functions.

The Function Notation Editor (MoNoWit) provides an effective visual record both of the basic time-space structures of a movement and of the evolution shapes of the parameters and also offers visual correspondence between the two kinds of movement representation (Function Notation and Symbolic Notation) by using the same time representation and angular scale system. This concept allows the transformation of Symbolic Notation into Function Notation and vice versa. The Function Notation Editor supports drawing, manipulation and generation of new graphs as the difference, sum or
average of existing ones. Functions can be assigned to body parts and movement parameters for generating or modifying MOVIE files. These facilities allow the creation and analysis of low level movement structures with a high degree of generality.

1.3 Symbolic Movement Notation (on Basic and Complex Levels)

The symbolic notation on its basic level describes the simultaneous changes of position in time for each active body part. The Symbolic Notation on a higher level is founded upon user-defined symbols, representing complex movement procedures.

![Image of the symbolic notation system]

Figure 1: The schematic representation of the internal communication structure of Motogramm/UNTIVUS

1.3.1 Symbolic Notation on Basic Level

The Symbolic Notation on its basic level is constructed to describe any type of human movement which can be represented in terms of changes of rotation angles and translations in a 3D space. This concept of movement representation provides a high degree of generality and reduces the number of symbol types.

Some investigations into symbolic motion description and editing have resulted in some significant results, e.g. LabanWriter by Scott Sutherland [Jackson, 92] or the Beneath Notation Editor by Ryman and Henry [Ryman, 90]. A basic function of all notation editors is that they can be used both to enter new notation sequences and to edit existing ones. In many cases the editor facilities and the internal data representations are highly dependent on the design and the graphics of the notation system used by the editor. The Symbolic Movement Notation Editor (Motogramm, founded upon Rajka Movement Notation [Husnyary et al, 92]) in the Motogramm system is an attempt to separate the movement data and its graphic representation as far as is possible. This concept facilitates a more autonomous treatment of the motion in itself within the program and data structures, which permits the development of more complex manipulative features. Another advantage of the above described concept, is the possibility to define new symb...

Interactive Performance 188

ICMC Proceedings 1994
symbols) and associate them with a procedure name, by selecting the name on the menu list. By resiting a shape, the actual duration of the associated procedure call will be evaluated. The duration of a procedure call indicates the performance time of the procedure as percent of its original duration. In case of a group composition, the library procedures can be called by different dancers with different timing, durations and mirroring.

Working with MetaDex, the user-only need click on the animation window to get an animated performance of the current score or to examine a single procedure.

2 Animation of Human Motion

A Metaphor is a tool in Motographon for visualization of human movement. It is not intended for rendering purposes, but to function as a tool for immediate checking of notated scores. However, Metaphor is able to translate MOVE files to be performed by LifeForms™ (Gaudet et al. [89]) which supports several rendering formats.

2.1 Anima II

ANIMA II is founded upon a parametric animation concept. It is characterized by objects defined by a set of parameters at each consecutive time step. The animation program creates frames (images) based on the assigned rotation and translation values as they are defined on a MOVE file. In the time span of the assigned duration value any number of in-between frames will be generated, limited only by the drawing rate of the actual computer system. In ANIMA II the user works with scenes consisting of dancers, a simple floor and a background. It is possible to view several scenes simultaneously in different windows. The camera control feature allows the user to change the viewpoint in 3D, by moving the mouse in the scene window, without interrupting the movements. For the execution of a choreography a movement control panel provides play, stop, replay and speed functions. The speed control is scaled in nemetos units.

Anima II is able to receive MIDI information—present time code for synchronization with a sound file or a CD player, and MIDI commands which can affect optional variables in a dancer’s MOVE data during the real time performance of the movements.

2.2 Life Forms™ and Motographon

Our research group has had an exchange of information with Tom Calvert and his research group at Simon Fraser University, which has resulted in a software link between our systems (Life Forms™ and Motographon). Life Forms™ is an advanced animation program based on facilities for interactive manipulation of outline figures. Motographon sequences created in Motographon can be converted to Life Forms™ format and—without limitations—Life Forms™ files can be converted to MOVE format.

3 Summary

The different notational systems, tools and applications in Motographon are intended to solve the problems emerging from the complex and hierarchical nature of the process of creation of movement structures. Motographon includes different notational systems and applications for different levels of movement description and manipulation. The real-time animation tools allow immediate visualization of human movements, synchronized with sound playback. Our experiments in applying Motographon to more extensive choreographic works and in dance teaching constitute an effort to establish a new, structure-oriented composition technique in the art of choreography, closely related to the development of the multimedia arts.

Acknowledgments

The above paper is based on research and technical development carried out within the framework of the Transmediaal Communication System NUNES (Enguy et al. [92]) by the Kineto-Auditory Communication Research Group (KACOR), mainly at the Royal Institute of Technology, Stockholm, with the support of the Bank of Sweden Tercentenary Foundation, of the Swedish Research Council for Engineering Sciences, of the Council for the Renewal of Undergraduate Education and of the National College of Dance, Sweden.

References


ICMCP proceedings 1994

Interactive Performance