MOTION EDITING AND REUSE TECHNIQUES AND THEIR ROLE IN STUDYING EVENTS BETWEEN A MACHINE AND ITS OPERATOR

Ovsei Volberg
Palladium Associates
201 S. Main Street, Suite 900
Ann Arbor, MI 48104
Tel: 734.971.1589
E-mail: ovsei@worldnet.att.net

Dean H. Ambrose
National Institute for Occupational Safety and Health
Pittsburgh Research Laboratory
Pittsburgh, PA 15236
Tel: 412.386.6650
E-mail: DAmbrose@cdc.gov

Keywords: motion editing, motion control, anthropometrics model, machine-operator hazards, computer animation.

ABSTRACT

Motion capture involves recording the position and global orientation of joint sensors of a real object, in most cases a real person performing some human activities. This information is usually recorded at uniformly spaced instances of time, or as it is often called frame-by-frame. Then the recorded motion data sets are processed and mapped into a skeleton hierarchy of a virtual, computer simulated human figure to control the motion of the virtual human in the computer simulation. In the first part of the paper we review several new techniques developed to facilitate the manipulation, noise reduction, storage and reuse of captured data, which have a potential to reduce the overall cost of motion simulation and improve its realism. In the second part of the paper we consider the real life problem of reducing a worker's risk from being hit by underground mining machinery in a confined space. We formulate a set of requirements to motion editing for this particular task and analyze the limitation of existing techniques.

INTRODUCTION

Starting from the mid 90's storage and reuse issues have become hot topics in many software-based technologies. In particular, the reuse issue motivated the introduction of Standard Template Library, Microsoft Component Object Model and design patterns, which are currently the "must-have" technologies in the diverse class of challenging applications, from e-Business applications to image and data processing programs. Human modeling and simulation (HMS) applications are not an exception. More than that, to be successful, contemporary HMS application development requires addressing the data storage and reuse issues through specific means in motion data processing. The main goal of all motion processing techniques is the creation of motion libraries and motion editing tools on top of them aimed to achieve the following:

- Keyframe Reduction.
- Motion Deformation - changing some parameters of motion.
- Retargeting Motion to new characters, for instance to human figures with anthropometrical characteristics differ from the person whose motions were recorded.
- Motion Cyclification - scaling the motion length in time while preserving the motion basic characteristics
- Motion Concatenation - connecting several pre-recorded and/or computer–generated motion curves with smooth transition between motion sequences.
- Motion Interpolation to create a motion "in between" two pre-recorded motions, for example, the creation of a fast walk motion from the recorded walk and run sequences.

We survey and analyze several modern techniques based on a B-spline interpolation, space-time minimization, and signal processing and physically based motion transformation. We use the problem of reducing worker's risk of being hit by underground mining machinery in bolting operations to discuss which modern motion editing techniques could be applicable to the problem and outline the possible approach to the creation of motion libraries and motion editing tools on top of them. We also describe a new type of motion warping based on the randomization of operator's behavior and equipment characteristics.

MOTION EDITING TECHNIQUES

The motion of some particular degree of freedom of articulated figures such as a human figure is usually represented by a collection of data points over time:

$$\{(t_1,x_1), (t_2,x_2), \dots , (t_n,x_n)\}, \qquad (1)$$

where x_i is the data value at time t_i. Using some curve fitting technique, usually some kind of interpolation or approximation, the sampled data set is converted to motion curve x(t).

The motion warping as an editing technique was introduced in [1]. Witkin and Popovic's method modifies the original motion curve x(t) by interactively turning the positions of pre-selected x-coordinates and by scaling and shifting the motion curve as whole. The interactive turning is done as time warp by introducing a set of pairs (t_j^*, t_j), each giving the time t_j^* to which the value originally associated with time tj should be displaced. As result the method could be described by two functions:

$$x^*(t) = a(t)x(t) + b(t), \qquad (2)$$
$$t^* = z(t). \qquad (3)$$

Function a(t) is used to scale the motion curve and b(t) is used to offset the center of scaling of x. The time-warping function z(t) is constructed as a Cardinal spline [1]. Thus, the pattern of the original motion curve is preserved, but the resulting curve satisfies the constraints of new key-frames.

The motion-displacement methods like [1] and [2] offer no explicit control over non-key frames. The lack of explicit control is addressed in space-time minimization methods [3-4] that consider entire motion curve in making change. To make those changes effectively the method [1] relies on three factors: the constraint formulation, which is simple enough to permit rapid solution; the fast method for solving this constraint problem; the new interaction technique to for specifying and visualizing changes to an entire motion. The method reformulates the motion transformation as a numerical constrained optimization problem:

$$\min g(x(t)) \text{ subject to } f(x) = c. \qquad (4)$$

Where x is a vector that represents the parameters of motion, g is a scalar function that represents the objective, and the scalar function f and the vector c describe constraints.

The edited motion curve m(t,x) is formulated as a sum of the initial motion curve $m_0(t)$ and a new motion curve d(t, x):

$$m(t,x) = m_0(t) + d(t,x). \qquad (5)$$

The algorithm allows a variety of interpolation curves to be used for m(t,x), such as linear and cubic interpolation, cubic B-splines and wavelet B-splines. In the interactive setting, the endpoint interpolating B-splines are recommended. There are two types of constraints in this method: constraints that encode specific aspects of the motion that should be preserved in the editing process and constraints that serve as handles to drive changes in to the motion. Most constraints are kinematic, that is they place restrictions on the configuration at a given instant. The other constraints are placed over a range of times. They are called variational constraints, because they describe a relationship over a continuous interval. In practice, the variational constraints are implemented as kinematic constraints placed at all instants within the specific time interval. Constraints fall into three categories: constraints on the character, such as joint angles; constraints that provide information about the initial motion, such as footholds and handholds; and constraints used to make adjustments, for instance interactive dragging that positions a point in a specific frame.

The objective function is formulated as

$$g(x) = 0.5x^T Sx, \qquad (6)$$

where S is a diagonal matrix of the parameter sensitivity functions. The method requires the ability to compute sensitivity functions as derivatives of the motion representation. Thus the problem is described as quadratic programming problem and can be solved by a class of algorithms known as sequential quadratic programming [3].

In [4] this method was applied to retargeting motion between articulated figures with identical structure but different segment length.

The space-time optimization approach, however, has several drawbacks, the most important of which are the complexity of space-time formulation and convergence difficulties.

An alternative approach to editing motion-captured data was offered in [5]. This method consists of four main stages. During **Character Simplification** stage a simplified model of human figure is created. The simplified model should have the minimal number of degrees of freedom necessary to capture the essence of input motion. The input motion is mapped to the simplified model. The next stage is called **Space-time Motion Fitting**. On this stage the space-time optimization problem is formulated to match the simplified character motion. The motion parameters and pose constraints could be changed during **Space-time Edit** stage. Finally, during **Motion Reconstruction** stage the changes in motion introduced by space-time edit could be remapped onto original motion to produce the final animation. The simplified model preserves the important dynamic characteristics of the original character. The character simplification and space-time motion fitting are not automated and require a significant amount of human intervention. The space-time edit and motion reconstruction are fully automated [5].

The paper [6] presents an algorithm for reparametrization of the one-dimensional motion signal in the discrete domain, according to user defined velocity function. A signal x(t) could be reparametrized by using a function p(t) as $x^{\sim} = x(p(t))$. The function p(t) changes the speed of the curve x(t). The authors of the paper [6] apply the derivative of the function p(t) to the collection of data points over time (1) representing joint angles obtained from human motion data capture to create the effects such slow-motion and accelerated-time. The derivative of the function p(t) is a velocity curve in [6].

The interpolation techniques [2, 7] use a database of motion characteristics either in frequency or time domain and interpolate between their parameters in order to produce new motions. The methods developed by Unuma et. al. [7] uses Fourier series to express trajectory of motion capture joint angles in the frequency domain. The low band filter is applied to keep only first modes (up to 3 modes) and their parameters. Once these parameters are obtained for different locomotion styles it is possible to interpolate from one set of parameters to another. Thus the new motion is completely defined by coefficients of interpolation and by two sets of original motions. For example, to make a human figure walk in more or less tired fashion, the interpolation between a normal and tired motion sets is used.

The main advantage of interpolation techniques is the low computational cost. The main drawback is the general assumption that the transition between parameters is linear or cubic and continuous in time, which is not always true.

An attempt to improve interpolation techniques is described in [8]. The architecture of motion editing system is presented. The system consists of simplified hierarchical representation of a human figure, a noise reduction module, a curve fitting module and a composition module. The use of magnetic and optical devices introduces some noise into motion data. The noise reduction is based on a Gaussian adaptive filter with a moving window. When there is no noise present, the filter behaves as a regular low-pass Gaussian filter. However, at the time t_j, where data changes abruptly, the filter ensures that the corresponding data point x_j contributes very little to the filtered data. The curve fitting technique is based on non-uniform B-splines and special heuristic procedures based on the geometry of the underlying data:

- Add knots in regions of high curvature.
- Add multiple knots to represent geometric discontinuous of the input data.
- Avoid large areas without knots.

The composition operations consist of unary and binary operations based on B-spline representation. As result motion warping, the same module could achieve motion interpolation and smooth transition between motion sequences.

The architecture for the motion capture based animation is described in [9,10]. The following three main objectives motivate the motion capture based system architecture [9]:

- To provide a set of tools for motion manipulation and analysis.
- To allow the production of high-quality complex animation, using reusable motion libraries.
- To compensate technological limitations of motion capture hardware.

The commercial ergonomic simulation and analysis software systems like Jack™ [10] lacks advanced motion editing capabilities, but provide a set of means like high level scripting languages and advanced animation system, which can be used by authors of specific simulation to design and implement motion editing application tailored for specific simulation purposes [11,12].

MOTION CAPTURE EDITING IN INJURY PREVENTION SIMULATION

The motion transformation techniques referenced in the previous section treat motions as collections of signals. In reality motions, especially collections of captured motion data sets already contain detailed information about actions they correspond to. Thus the rule-based intelligent mechanism can be formulated on properties of the action, such as targets/goals to enrich techniques based on signal processing. In [11] the novel motion editing approach based on the randomization of captured data and simulation parameters was introduced.

Injuries to operators of underground coal mining equipment have led an investigation of safe velocities of a roof bolter boom arm at the National Institute for Occupational Safety and Health (NIOSH), Pittsburgh Research Laboratory (PRL). Because empirical data cannot be collected from the incidents and laboratory experimentation is too complex and dangerous, a computer-based, three-dimensional solid model simulation approach was used as the primary means to gather data on mishaps. In the computer model, mishap means that two or more object properties interacting. Consequently, hazardous conditions were analyzed in virtual environments using collision detection.

The roof bolter computer model uses basic bolting motions created from training videos, in-mine observations and videos and critiques from bolter manufacturers and experts. Randomness was incorporated into the model to add realism and accuracy. The simulation generates and collects collision data between the machine and its operator and records the results such as, the kneeling or standing postures, choice of risk behavior, anthropology, machine's appendage velocity and mine environment. Jack simulation software was the simulation tool chosen to develop the computer model; it is a human-centric visual simulation software package and the software's architecture lets users extend its simulation functionality.

The uncertainty and variability inherent in the bolting tasks were incorporated into the model effectively to find out the likelihood of an operator being injured. Researchers accomplish a realistic representation of the operator's motions and behaviors typically in actual underground coal mine roof bolting practice by developing a motion warping technique to create random motions with the integration of captured motion data. Thus, for a machine and operator, the operator's various risk behaviors, motions for each risk behavior, and motion paths associated with each motion behaviors have some degree of randomness.

The new motion editing technique incorporates random motion in such a way that individual paths will differ slightly though the basic motions will look very similar without distortion. To figure out the range of these differences, laboratory motion tests were conducted. Researchers obtained data using a bolter machine's boom arm mock up and experienced roof bolter operators. The operators were asked to do prescribed motions with the mock up that simulated actual practice. The motions were recorded and data analysis provided standard deviation (SD) values for the operator's start positions and movements on hands, legs, and the head for all testing scenarios that reflect 25-, 55- or 92-percentile operators. Motion interpolation could enhance this process of the new technique by combining captured motion data with the basic and previous created motions.

The random-motion warping technique works through code was that developed to extend the main menus of Jack software version 1.2. The code provided unique selections that allow the end-user to apply modifications to basic motions before running the simulation. A random decision within the code is made regarding a risk behavior involving hands on the boom arm and on the drill steel or bolt. Operator risk behaviors, major model input parameters, were based on statistics of machine and human actions that could cause injuries or fatalities in a bolter's workspace [13]. Depending on the risk behavior, the motion warping technique creates for Jack's motion system [14] various operators' motions. The operator size and appendage test speeds are preset in the virtual environment prior to running simulations. Motion cyclification could enhance the new technique by varying the motion and their length in time to reflect object speed changes such as in the boom arm. Also, the use of retargeting motion could allow operator anthropometrical changes along with their behaviors.

Jack's software manipulation process defines how the model's operator is to achieve the final posture for the whole body or head, back, hand, arm or leg. The motion that the operator goes through to achieve a final posture is described only through Jack's motion system. For example, the manipulation values for xyz-orientation angles and xyz-positional coordinates define the final posture position of the operator. Then the human motion system's algorithm generates and animates the motion-path to achieve this final posture. The motion system is neither completely discrete nor completely continuous; therefore, it blends very well with the new random-motion warping technique and leads to aspects of both discrete-event and continuous simulation. A unique, combined discrete-continuous simulation is accomplished by built-in random manipulation values within the model before transformed into a motion-path by the human motion system. Jack's motion system would reflect the motion warping in a motion-path as defined by these manipulation values.

The manipulation values produced by the random motion technique could be described by the following, using positional "x" as example:

Motion (positional "x") $\Rightarrow$ $(X - N_1) *$ [a random number from zero to N_2]

Where X is the basic positional value, N_1 is the SD value for scenario, which is obtained from a look up tables within the code, and N_2 is the SD value multiplied by two.

The new motion warping technique has several rules that must be applied to maintain the integrity of the basic bolting motions. Before applying the rules, all of the basic bolting motions in a scenario must be well defined and understood as to resulting direction. The rules decide how the much editing is done to each of the motion's manipulation values for xyz-orientation angles and xyz-positional coordinates see Chart 1. Other possible combinations were omitted from the chart, because they were not needed. Motion deformation technique would be a good candidate for trying to incorporate rules that apply to changing parameters of motions prior to creating them.

Hazardous conditions of the operator are realized in virtual environments using collision detection. The operator's vision as to what is seen during the bolting task becomes crucial when figuring out what collisions between the machine and operator could have been avoided. The operator's response time as to how quickly he could move out of the way is needed. Data was collected on operator response times. Data was collected on human subjects that determined viewing cones of humans in low light conditions found in mining environments. Collision detection, distances calculations between reference points on an optimal viewing cone, and the object wanting to track are used to determine what collision-could-have-been-avoided. Currently the resolve for collision avoidance is to compare simulation time that a collision occurs to the vision information of the same time that defines whether the object, such as the boom arm, was seen or not. By manually analyzing data from the response times, collision detections and vision information, researchers obtain answers for collision avoidance issues.

Vision-motion editing scheme should be investigated with the possible use of space-time-edit so to automate the collection and analysis of collision avoidance issues. Using random-motion warping technique to change the duration length of the motions was not used because variances in the length of time to do bolting tasks were insignificant. The use of the random technique on the collision detection threshold was considered for response times; however, vision and collision avoidance issues could not be experimented with because of limitations with the software's motion system such as no editing allowed during system activation.

The most difficult use of the random technique is setting up modules of Lisp code for each motion of the operator that we chose to randomize. Also, much effort is needed in developing SD lookup tables. SD tables represent, in our case, mine seam height, operator kneeling or standing postures, and operator's anthropometrics. Additional effort is needed in developing or duplicating different model environments having appropriate motions that depict desired test speeds.

CONCLUSIONS

Random technique could be enhanced in part by several main motion editing processing techniques such as motion deformation, re-targeting motion, motion cyclification and motion concatenation. Editing motion captured data share similar stages of development and shortcomings, such as the simplified model preserves the important dynamic characteristics of the original character. Also, characters-simplification and space-time system both are not automated and requires much manual work. The times in a space-time-edit would enhance random technique by defining operators' response times for motions that removes the operator from collisions or possible collisions. Interpolation techniques could be used in the random techniques to help improve the use of look up tables and the random technique process that combines captured data prior to creating motions. When overlaying editing features over any software, the challenge will be to ensure compatibility to the software's motion system, reduce end-user interactions, and reduce the need to write code by auto-writing it.

As a minimum, we need motion editing interfaces that allows the user to enter initial setup parameters that result, for example, in the following: with some degree of randomness have the operator walk into an environment containing several objects (machine, tools, etc.), ready itself into a posture to begin the work task, looks at an object and successfully reaches and grasps it. The random events could include the operator's posture while working, the object(s) in the environment that the operator uses to do the job and the hand for the reach and grasp event, to have the operator behave randomly, and vary the machine appendage speeds. Finally, run a pre-processor discrete event simulator to create the setup and allow the simulation to run repeatedly without intervention.

Chart 1 – Random Motion Technique's Rules *

		OPERATOR'S MOTION						
		Basic	Prominent vertical direction	Vertical with z direction	Vertical with x direction	Point to point	Leaning forward & backward	Starting position
DIRECTION	Orientation angles	x y z	x y z	x y z	x y z	x y z	(random x) y (random z)	(random x) y (random z)
	Positional coordinates	x y z	x (random y) z	x (random y) (random z)	(random x) (random y) z	(random x) (random y) (random z)	x y z	(random x) y (random z)

*The Common Lisp facility for generating pseudo-random numbers has been carefully defined to make its use portable. The function, random *number*, accepts a positive number n and returns a number of the same kind between zero (inclusive) and n (exclusive). An approximately uniform choice distribution is used [17].

REFERENCES

[1] Witkin, A. and Popovic, Z. 1995. "Motion Warping." In *Proceedings of SIGGRAPH 95*, 105-108.

[2] Bruderlin, A. and Williams, L. 1995. "Motion signal processing" In *Proceedings of SIGGRAPH 95*, 97-104.

[3] Gleicher, M. 1997. "Motion Editing with Space-time Constraints". In *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, 136-148.

[4] Gleicher, M. 1998. "Retargeting motion to new characters". In *Proceedings of SIGGRAPH 98*, 33-42.

[5] Popovic, Z. and Witkin, A. 1999. "Physically-based motion transformation". In *Proceedings of SIGGRAPH 99*, 11-20.

[6] Silva, F., Velho, L. and Gomes, J. 1998. "Motion Reparametrization". In *EUROGRAPHICS Technical Note*. 1.5.1-15.4.

[7] Unuma, M., Anjyo, K. and R.Takeuchi 1995. "Fourier principles for emotion-based human figure animation". In *Proceedings of SIGGRAPH 95*, 91-96.

[8] S. Sudarsky. 1999. "Manipulation and Reuse of Motion Data", PhD thesis, Texas A&M University.

[9] Silva, F., Velho, L. Cavalcanti, P., Gomes, J. 1997. Architecture for Motion Capture Based Animation". In *Proceedings of SIBGRAPI'97, X Brazilian Symposium of Computer Graphics and Image Processing*. 49-56. IEEE.

[10] Badler N., Phillips C., and B. Webber, B. Simulating Humans: Computer Graphics Animation and Control. Oxford University Press, New York, NY, 1993.

[11] Ambrose, D.H. 2001. "Random Motion Capture Model for Studying Events Between a Machine and its Operator". *Proceedings of Business and Industrial Simulation Symposium, ASTC'2001*. 127-134.

[12] Volberg, O. 2000. "Robotic inverse dynamics for simulating virtual humans in virtual manufacturing and design". *Proceedings of Business and Industrial Simulation Symposium, ASTC'2000*. 64-69.

[13] Ambrose D.H. 2000. A Simulation Approach Analyzing Random Motion Events Between a Machine and its Operator. *Proceedings of the 2000 SAE Digital Human Modeling for Design & Engr*, nr. 2000-01-2160, pp 10.

[14] Bartels J.R., Ambrose D.H., Wang R. 2001. Verification and Validation of Roof Bolter Simulation Models for Studying Events Between a Machine and its Operator. *Proceedings of the 2001 SAE Digital Human Modeling for Design & Engr*, nr. 2001-01-2088, pp 13.

[15] Klishis, M. J., et al, 1993. Coal Mine Injury Analysis: A Model for Reduction Through Training, Mining Extension Service, West Virginia University, Morgantown, WV. August. pp. 220.

[16] Azuola F., Badler N. I., Ho P., Huh S., Kokkevis E., Ting B., 1996. Jack Validation Study: Final Report. University of Pennsylvania, Center for Human Modeling and Simulation, Philadelphia, PA. Grant No. DAMD 17-94-J-4486-P50002, pp. 138.

[17] Steele, G. L. Jr. 1990. Common Lisp, The Language. Digital Press, ISB 1-55558-041-6.

BIOGRAPHY

Dr. Ovsei Volberg earned his Bachelor of Science in Computational Physics/Computer Information Systems and his Ph.D. in Electrical Engineering (Systems) from Technical University, St. Petersburg, Russia (Formerly Leningrad Polytechnic Institute). Computer simulation of dynamic and control systems, Virtual Reality application development and robotics are his main research interests.

Dr. Volberg worked in robotics research. At the Russian Academy of Science he was responsible for design and implementation of the sophisticated algorithms of a four-leg mobile robot, its numerical simulation and control. Another area of his research activity was in the area of the robotic path-tracking problem subject to parameter uncertainty and external disturbances. He proposed the new sliding-mode-control-theory-based scheme. After his arrival in the USA in 1991, Dr.

Volberg started to work in industrial robotics as a software developer for Tecnomatics Technologies. Dr. Volberg was one of the principal developers of ROBCAD-based off-line programming (OLP) software. Dr. Volberg designed and implemented multiple OLP packages for Fanuc, Nachi, Panasonic, ABB and Kawasaki industrial robots. His most recent research interest is Digital Human Simulation. This includes the physics-based digital-prototyping of biomechanical systems and the development of simulation and animation software, including Virtual Reality applications, for human motion analysis. Dr. Volberg was one of the principal developers of the non-graphical toolkit inside Classic (Transom) Jack™ Human Simulation software - a sophisticated modeling and animation product for ergonomic simulation. He also implemented several collision detection routines and ergonomic analysis tools for Classic Jack™. Dr. Volberg participated in the design and development of several Virtual Reality applications employing CyberGlove and MotionStar motion capture devices. Dr. Volberg took an active part in the deployment of the Jack™ Human Simulation software applications in the automotive industry (Ford Motors, GM, Volvo, Toyota, Honda Motors, etc.), the aerospace industry (Lockheed, Boeing, Bombardier, etc.) and the US Army (TACOM and Soldier and Biological Chemical Command).

Dean Ambrose has been working for the Federal government for more than thirty years in both military and civilian positions. He is a retired Lt. Colonel from the Air Force with 10-years of activity duty and 17-years of reserve duty. In 1978, his civilian service began and he continued his electrical and computer engineering career with the Department of Interior's Bureau of Mines. Now with National Institute for Occupational Safety and Health (NIOSH), he conducts research that leads to discovering techniques that improve health and safety in the workplace for mining, agricultural and construction industries. For the past ten years, Dean has been enjoying his research interest, which is applying modeling and simulation technologies to investigate control interventions that reduce hazardous conditions and events in the workplace, in particular for underground and surface mines. He has several publications that address the use of modeling for machine safety research. Dean is a 2001 recipient of CDC/ASTDR Honor Award in Epidemiology and Laboratory Research for his contributions in roof bolter safety through research on the Human Factors Design for Machine Safety Project of which he is the Principle Investigator.