DEVS-based Modeling of a Human Motion Data Synthesis System based on Motion Capture Data

Seyed Ali Etemad and Gabriel Wainer
Department of Systems and Computer Engineering
Carleton University
1125 Colonel By Drive
Ottawa, ON. K1S-5B6 Canada
{etemad, gwainer}@sce.carleton.ca

Keywords: DEVS, human motion, motion capture data, synthesis, body.

Abstract

This paper presents a DEVS-based model for a human motion data synthesis system. The model includes some major components of the human motion control system such as brain, spinal cord, and body. The system is designed to produce the output data in the form of motion capture data. The proposed model and the output data produced can be widely applied in areas such as robotics and multimedia (animation). The simulations carried out illustrate the performance of such a model which is capable of accurate mimicking of the human motion control system. The model is capable of taking into account minute features either related to actor characteristics, or even noise. In the end, we have shown that discrete event modeling is a very suitable means of describing human motion control and synthesis systems.

1. INTRODUCTION

Human motion is an important field of study for multimedia, robotics, and biomedical experts. Various applications such as security, surveillance, computer games, animation, robotics, physiology and etc have resulted in a great deal of attention being drawn to this topic. While some of the applications require classification of such data (which has extensively been addressed in different literature), a comprehensive model for synthesis and control of human motion data are an important issue yet to be fully resolved [1].

The human body by nature is a complex system (from a physiological point of view) and possesses large number of joints and different DOFs in different parts of the body (from a mechanical point of view), thus the human body motion is often difficult to describe as a single system. As a result of this complexity and the demand for more precise and capable models, the field is open for further research.

Two different themes are present when dealing with human motion [2]. The first theme is called the primary theme. This theme is composed of the features which create and describe different classes of every day actions. Actions such as walking, running, jumping, waving, sitting down, and even more complex actions such as dancing or fighting are all primary themes (or temporal combination of several primary themes). The other theme is called the secondary theme. These themes are accumulated on the primary theme and add to the action, personality and style, or sometimes provide additional information regarding some characteristics of the actor. Characteristics such as age, gender, energy, mood, health, and many more, are represented and pointed out by the secondary themes.

Different means of recording motion data are available. Mechanical, inertial, magnetic, and camera based systems are all capable of documenting the motion in some form. One of the most common forms of data recording is by means of camera-based systems. Motion capture systems are very accurate camera-based systems which we have employed to capture the human motion data and build the model based on this type of data. Motion capture data are extensively employed for animation and even robotics.

DEVS [3], in general, are systems with discrete events as outputs. While for some systems, discrete time approaches are common, discrete event techniques are not as widely practiced. The proposed model in this research is based on a discrete-event framework.

In this research, we have proposed and tested a DEVS-based technique for modeling and simulating a human motion data synthesis and control system. We have shown that DEVS are appropriate for modeling human motion based on motion capture data. The constructed model is trained to synthesize three classes of human action and produce the desired action (motion capture) data for a given initial pose. The synthesized output is animated and evaluated. Furthermore, secondary themes of the motion sequences are also tweaked and controlled using this model.

2. RELATED WORK

Various modeling techniques have been employed to describe human motion with most possible precision. Park et al [4] propose and construct memory based models. These models perform more resembling databases which also enables learning of new simple motor skills based on the database.

In [5] Zheng and Suezaki develop a simple graphical model which specific keyframes can be provided to the model and the motion between the frames is interpolated.

Ahmad et al [6] utilize motion captured images of motion. The aim of their research has been to illustrate how the type of a specific model influences its performance. They demonstrate that multi-variable non-linear models are most suitable to represent human motion (using the specific type of data).

Suleiman et al [7] utilize motion capture data and propose that the motion of different sections of the body is related. Their non-linear model employs the motion curves of the pelvis and produces entire body motion for different actions.

Regular cameras and specific markers is employed in [8] by Ahmad et al, for capturing human motion data. The exact motion data is extracted from the images, and the data modeled by on-linear models of kinematics.

Various modeling approaches have also been proposed with the aim of recognition and classification of human motion data. A.I.-based techniques are quite popular for this purpose [9 – 15]. Fannti et al [9] employ probabilistic models for recognition of human motion. They utilize different variables through their model as it is learned by an unsupervised manner.

Hidden Markov models (HMM) -a form of Bayesian networks, are very popular tools for modeling human motion data [10 – 12]. In [10] Etemad and Arya utilize these models for classification of motion capture data for both primary and secondary themes. Different forms of Bayesian networks have also been used to model human motion data. Stoll and Ohya [11] have employed ordinary hidden Markov models and visual data to compose a system for recognition of six classes of action. In [12], Pavlovic and Rehg have compared the common HMM to switching linear dynamic systems (SLDSs), -another form of Bayesian networks, for recognition of human motion.

ANN has also been employed for modeling of motion data. Simple back-propagation networks [13] as well as more complex learning models such as recurrent [14] and resilient [15] have also been utilized. In [13] Etemad et al train neural networks in the form of anticipators using optical features. Ogihara and Yamazaki [14] model the nervous system capable of producing motion signals using recurrent neural networks. Finally Etemad et al [15] employ motion capture data for training resilient neural networks as anticipators capable of classifying and synthesizing human motion data.

In the end, some rather creative and interesting approaches have also been taken towards tackling the problem. For instance, the inverted pendulum model (IPM) has been utilized by Tang et al [16] to model and simulate human motion.

The general trend when applying A.I.-based techniques for modeling (especially for human motion) has been the fact that these systems have been treated as black boxes which tend to learn and describe human motion. For all types of modeling, A.I.-based or not, the common element in the mentioned literature has been the factor of event continuity and often temporal continuity. These two characteristics, however, are not imperative for describing human motion data. Also, despite its importance and usefulness, modeling based on motion capture data has not been popular.

3. BACKGROUND

As mention in section 1, two of the most important aspects of this research are event discontinuity and utilization of motion capture data. In this section, the concepts of DEVS (discrete event variable systems) and motion capture data are presented as an inclusive understanding of the two topics is essential for the aim of this research.

3.1. DEVS

DEVS are one of the major categories of modeling techniques based on system dynamics [3]. They are widely employed for modeling and simulation of systems with discrete events. A key fact regarding DEVS is the independence of the simulator and the constructed models, thus making the frame work practical and interoperable.

Different systems are modeled using a hierarchy of singular models called atomic models. A set of two or more interconnected atomic models form a coupled model. Coupled models are used to model some aspect of the system which is composed of sub-models itself, usually performing more complex tasks.

As atomic models are the building blocks of DEVS, it is quite essential to provide a description of atomic models and describe their functionality. Figure 1 illustrates a descriptive sketch of an atomic model. An atomic model is specified by equation 1.

$$\boldsymbol{M = \langle X, Y, S, \delta_{int}, \delta_{ext}, \lambda, ta \rangle} \tag{1}$$

X and *Y* are the set of *input* and *output* events respectively. *S* is the set if sequential *states* which their durations are determined by *ta* which is the *time advance* function. The atomic model is connected to other models via the *input ports* (*x*) and *output ports* (*y*) which carry the events. λ is the *output* function of the atomic model which sends the outputs which have been produced as a result of the execution of the model. δ_{ext} and δ_{int} are the *external* and *internal* state transition functions respectively. The former determines the reaction of the model towards external events while the latter produces a local state change successive to production of the outputs.

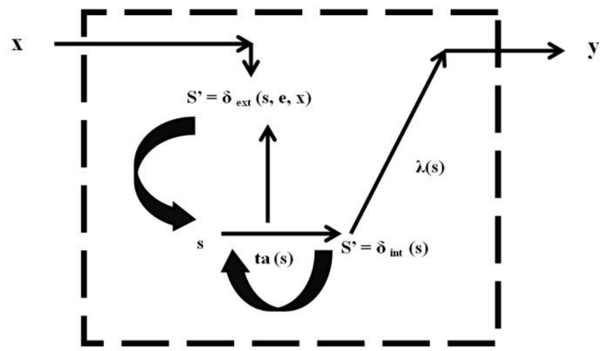

Figure 1. DEVS atomic model semantic

3.2. Motion capture data

Motion capture data obtained by means of a six-camera Vicon system in the School of Information Technology, Carleton University, have been used for this research. We have asked some actors (students) to perform the required basic action sequences. Figure 2 illustrates a caption of a motion capture session.

Figure 2. Motion capture session

The motion capture data come in the form of equation 2, where D_i are the Cartesian values for the hip marker in 3D space with respect to the calibration origin and Θ_i are the rotation angles in degrees for each marker. There are m rows, denoting m frames.

$$A = \begin{bmatrix} D_1 & , & \Theta_1 \\ D_2 & , & \Theta_2 \\ \vdots & & \vdots \\ D_m & , & \Theta_m \end{bmatrix} \tag{2}$$

The positioning marker is presented in Figure 3 by the marker connecting the two legs (orange marker), which corresponds to the marker placed on the hip. This marker provides the Cartesian measures for locating the actor in each frame of action. Figure 3 also shows the axis of one of the joints. Each marker on the body possesses its own frame of reference similar to those shown for the left leg. Figure 4 illustrates a real scene of the markers placed on the specific motion capture suit.

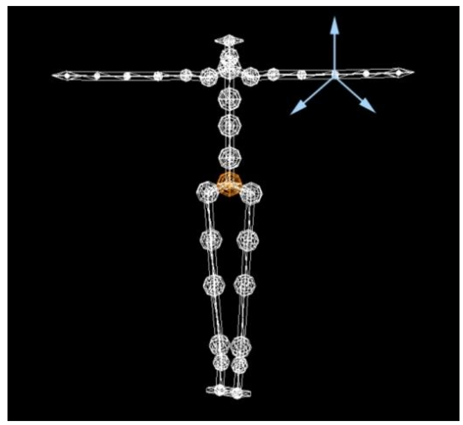

Figure 3. Marker orientation and the axis of one of the markers

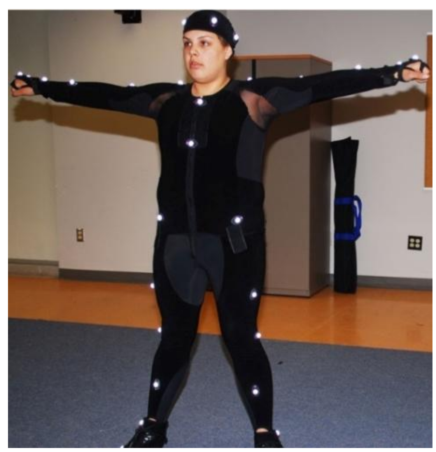

Figure 4. Marker orientation in real motion capture session

Figure 5 illustrates a walking sequence for 45 frames where snapshots of each 5th frame have been presented. Also the movement of 5 of the markers is illustrated using different colors.

The rotation values matrix for i^{th} marker for frames 1 to m is as follows, where $\theta_j^{x_i}$ denotes the rotation value of the x coordinate in space, related to i^{th} marker of the j^{th} frame:

$$\overline{\theta}^i = \begin{bmatrix} \theta_1^{x_i}, \theta_1^{y_i}, \theta_1^{z_i} \\ \theta_2^{x_i}, \theta_2^{y_i}, \theta_2^{z_i} \\ \vdots \quad \vdots \quad \vdots \\ \theta_m^{x_i}, \theta_m^{y_i}, \theta_m^{z_i} \end{bmatrix} \quad (3)$$

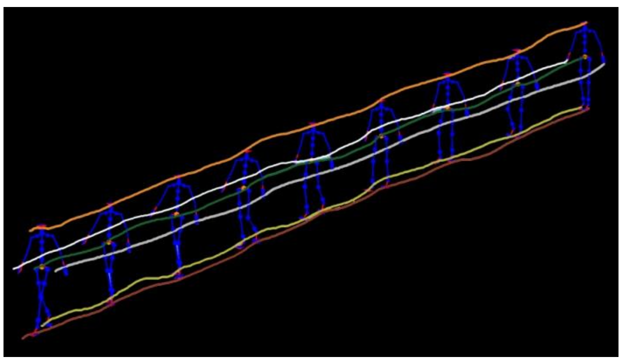

Figure 5. A walk cycle and the movement of 5 of the markers

In equation 4, the complete angular rotation matrix of m frames and n markers is presented.

$$\begin{bmatrix} \Theta_1 \\ \Theta_2 \\ \vdots \\ \Theta_m \end{bmatrix} = \begin{bmatrix} \left(\theta_1^{x_1}, \theta_1^{y_1}, \theta_1^{z_1}\right), \left(\theta_1^{x_2}, \theta_1^{y_2}, \theta_1^{z_2}\right), \ldots, \left(\theta_1^{x_n}, \theta_1^{y_n}, \theta_1^{z_n}\right) \\ \left(\theta_2^{x_1}, \theta_2^{y_1}, \theta_2^{z_1}\right), \left(\theta_2^{x_2}, \theta_2^{y_2}, \theta_2^{z_2}\right), \ldots, \left(\theta_2^{x_n}, \theta_2^{y_n}, \theta_2^{z_n}\right) \\ \vdots \qquad \vdots \qquad \vdots \qquad \vdots \\ \left(\theta_m^{x_1}, \theta_m^{y_1}, \theta_m^{z_1}\right), \left(\theta_m^{x_2}, \theta_m^{y_2}, \theta_m^{z_2}\right), \ldots, \left(\theta_m^{x_n}, \theta_m^{y_n}, \theta_m^{z_n}\right) \end{bmatrix} \quad (4)$$

The related data for the hip positioning marker for frames 1 to m is shown by D. In equation 5, d_i^x represents the value of the x coordinate of the distance of the hip marker with respect to origin for frame j.

$$\begin{bmatrix} D_1 \\ D_2 \\ \vdots \\ D_m \end{bmatrix} = \begin{bmatrix} d_1^x, d_1^y, d_1^z \\ d_2^x, d_2^y, d_2^z \\ \vdots \quad \vdots \quad \vdots \\ d_m^x, d_m^y, d_m^z \end{bmatrix} \quad (5)$$

The final form of the data is presented by equation 6 where $\theta_i^{x_j}$ represents the rotation values of the x coordinate of the j^{th} marker for the i^{th} frame and d_i^x represents the position of the x coordinate of the hip marker for the i^{th} frame.

$$A = \begin{bmatrix} \left(d_1^x, d_1^y, d_1^z\right) & \left(\theta_1^{x_1}, \theta_1^{y_1}, \theta_1^{z_1}, \ldots, \theta_1^{x_n}, \theta_1^{y_n}, \theta_1^{z_n}\right) \\ \left(d_2^x, d_2^y, d_2^z\right) & \left(\theta_2^{x_1}, \theta_2^{y_1}, \theta_2^{z_1}, \ldots, \theta_2^{x_n}, \theta_2^{y_n}, \theta_2^{z_n}\right) \\ \vdots & \vdots \\ \left(d_m^x, d_m^y, d_m^z\right) & \left(\theta_m^{x_1}, \theta_m^{y_1}, \theta_m^{z_1}, \ldots, \theta_m^{x_n}, \theta_m^{y_n}, \theta_m^{z_n}\right) \end{bmatrix} \quad (6)$$

The presented matrices in the form of (6) are documented and recorded by the motion capture system. The file which records the matrices are called BVH files. These files are composed of two sections. The first section is what we refer to as the header section. The headers section describes the orientation of the markers and their initial offset from the hip marker in hierarchical format. The second section is the body. The body is a matrix similar to that shown in equation (6). Some extra information such as frame rate is presented between the header and the body of BVH files. After a precise and complicated calibration process, the required BVH files are recorded from which valuable information is extracted. The information can then be utilized to create different systems for describing human motion.

To make the data more suitable for DEVS, the data are quantized to create discrete signals. This is illustrated in Figure 6 where the blue signal represents the quantized discrete data and the red signal is the original continuous data.

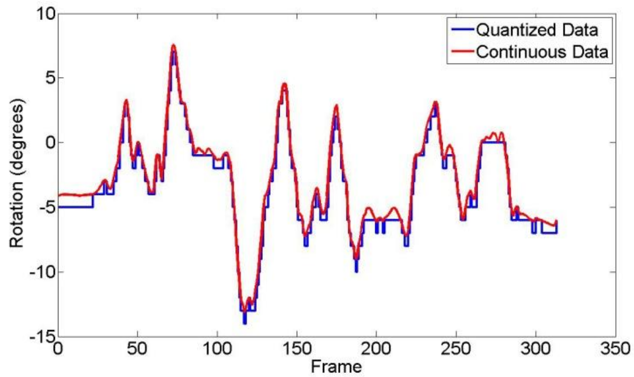

Figure 6. Continuous vs. Discrete data

4. MODELING AND SIMULATION

The key to design and construction of a precise human motion data synthesis system is taking into account all the major components of the human body. The proposed model for this research is composed of three main sections. The first is the brain. The brain is the section responsible for learning and synthesizing all the motion data. Spinal cord is another section of the system which acts as the data transmission and distribution route from the brain to the other sections of the body. The spinal cord carries the messages produced by the brain to muscles responsible for human motion. Finally the third section of this model is the body. The body, in this case, represents only the markers placed on the motion capture suit. Figure 7 illustrates the overall model.

We will first describe the setup and the functionality of the *brain*. The brain contains three ports. Input ports (*X*) are *input* and *current*. The output port (*Y*) is called *brainout*. A minimum of two states (*S*) are required for this model. The first is called *rcv_in* and the second is called *rcv_stat*. The brain basically contains some sort of an algorithm for synthesis of motion data, either using database or even an A.I.-based system in the form of a black box. The research here aims at demonstrating a proof of concept that DEVS bring about a suitable platform for describing human motion and is not concerned with the type of motion synthesis algorithm.

Four different actions are projected to be tested using the proposed method. The actions are: walking (masculine and feminine), jumping, running, and kicking. Each action is embedded in the brain as assumed to have been learned through some algorithm. They are named *walk1*, *walk_fem*, *jump*, *run*, and *kick*. To facilitate the functionality of the brain, external input files containing the motion data are provided. The initial motion data is embedded in a file called *matrix.in*. Other action classes and types are provided in another file called *other.in*. Upon user request, the data from the *matrix* file can be substituted by data from the *other* file.

The input to the brain is in the form of a 1 x 81 vector. This vector is the initial pose for the action which is currently triggered inside the brain. For instance if the action walk1 (masculine walk) is triggered, the input can be any of the existing poses of the masculine walk sequence. The brain is initially in passive mode and in the *rcv_in* state. Upon receiving the intended pose, the state will change to *rcv_stat* which is in passive mode.

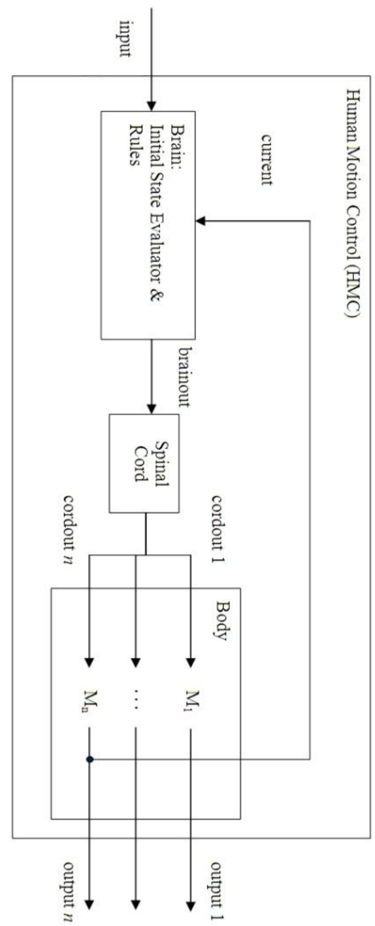

Figure 7. Proposed HMC model

The *rcv_stat* state is the state when the brain is waiting for acknowledgment through the feedback link from the body parts that the action is correctly in progress. This is modeled based on the characteristics of the human body where the brain can override any existing action-in-progress upon necessity. Therefore the brain in the model will not order further progress of the action prior to proper feedback from the body.

Once the brain receives the proper information from the *current* port that the sequence has succeeded a bit (1 frame), it will compute and send out the data for the consecutive bit (frame) of the action.

The spinal cord is connected to the brain. The brain sends out the data to the spinal cord. The *cord* model, similar to the spinal cord of the human body, has several functionalities. The first and the most important is the distribution of signals received from the brain to different sections of the body. The spinal cord receives the signals and provides the proper links between the brain and the different parts of the body.

The cord model has one input port which is connected to the *brainout* port of the brain. The *cord* receives the actions signals through this port and distributed them via the 81 output ports called *cordout1* to *cordout81*.

2 states are foreseen for the cord model. The first is called *wait_row*. In this state which is a passive phase, the spinal cord waits for arrival of new data from the brain. Once it receives the data, it goes into the *output_row* state which is responsible for locating and distributing the body parts which each signal must go to. After this task is accomplished, the model goes back to the *wait_row* state.

Another important property of the spinal cord is its ability to affect the signals which it receives from the brain. Different motor diseases such as arthritis can be caused by damage to the spinal cord. This property must be taken into account in the model. The atomic model, *cord*, possesses the ability to include a desired amount of noise using the suitable mathematical function. All this is done through the output function of the model. The results of a test for noise addition are presented in section 5.

The next model is the body. The body has 81 input ports which represent the different nerves connecting the spinal cord to the body for the transfer of motion signals. They are named *bodyin1* to *bodyin81*. Two output ports are designed for this model. The output ports are called *current* and *output*. The port *current* is what was referred to in the brain model by the same name. This port sends a signal to the brain stating that the current frame of the action is carried out successfully. The other output port, *output*, presents the pose of the body in each frame.

Similar to the model of the spinal cord, the model of the *body* operates in two states called *wait_row* and *output_row*. The first state acts as a passive phase waiting for new data. Once it receives the data, the body goes to the desired pose and the current feedback link is activated to inform the brain of completion of this frame.

Also similar to the spinal cord, noise can be added inside the body. Noise inside the body can be a representation of muscular diseases which can cause motor disabilities. The noise, once again, can be added in any desired fashion through the output function of the model.

The time values of the each of the actions in the models are selected carefully to resemble the performance of the body. The time delay for the brain to send out the data for each frame of action is 40 milliseconds. In reality, human motion is continuous, however, a motion sequence with a resolution of 25 frames per second or higher, is perceived by the human eye as one continuous motion. 25 frames of motion per second require 40 milliseconds of delay between consecutive frames. Other components of the system, which are basically responsible for conveying the movement signals, are assigned the minimum delay of 1 millisecond.

The next step is to provide different action classes and styles (primary and secondary themes) to the system and test and animate the outcome. The effect of noise must also be demonstrated.

5. RESULTS AND DISCUSSIONS

Once the models are created and running properly, the model is tested through different action classes and secondary themes. The output results are fed to a software called BVHacker. BVHacker is a software which can animate motion capture data. All the results in this research are animated and evaluated using this software, and required screenshots of the sequences will be provided.

To test a motion data system, different primary and secondary themes, as well as different possible first action frames must be employed. The selection of actions is carried out such that it would include a wide range of different motion signals. Walking and running have been chosen to illustrate the ability of the system to distinguish actions which are quite similar except for the speed at which the two primary themes take place. Both these actions are actions in which the location of the body has changed by the end of the sequence. Kicking is chosen to resemble an in-place action with one moving part of the body. Jumping is selected to resemble another in-place action, this time with the entire body moving and ending up in the same starting pose. For secondary themes, masculine and feminine themes for walking are tested.

The initial pose for an action is provided to the system. Based on the type of action which the model is set to synthesize, the brain must produce and output the consecutive frames 40 milliseconds after it has been confirmed via the *current* port that the following poses have been constructed successfully. Figure 8 illustrates the 5 actions produced by the system when the pose of frame 1 is provided to the system via the *input*. Also to test the system, for masculine walk, the poses for frames 6 and 20 are also provided to the system. As expected, the successive frame are produced and outputted through the body. The system is designed to return null poses (frames consisting of zeros only) if a match for the initial pose is not found.

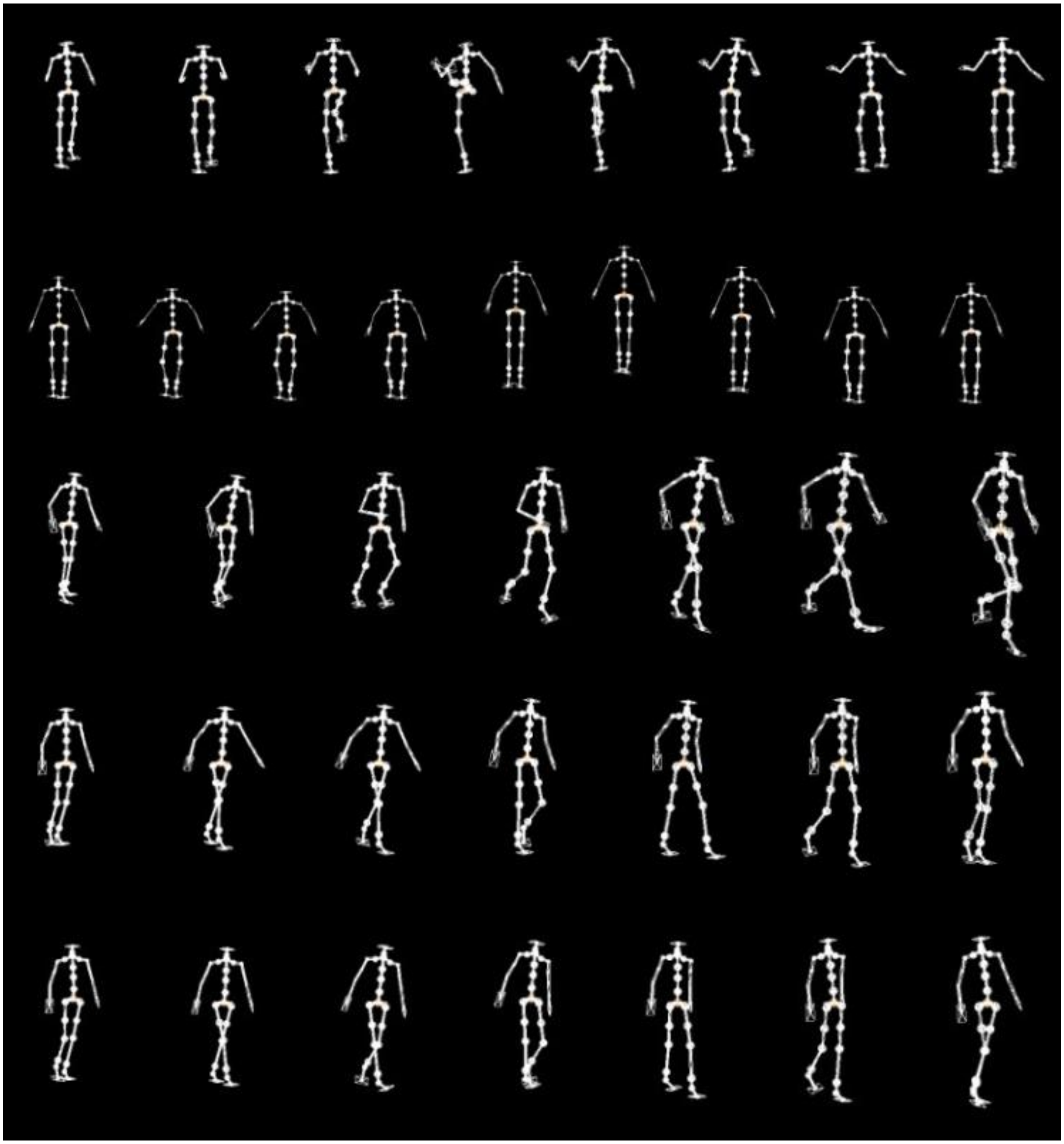

Figure 8. From top to bottom: Kicking, Jumping, Running, Masculine Walk, and Feminine Walk

Based on [17], different secondary themes are produced in the brain. As clearly shown in Figure 8 the bottom two sequences clearly illustrate the difference between a masculine and feminine walk. While the masculine walk appears to have more energy, the feminine walk appears to have slightly more speed. Also the movement of the arms in masculine walk is greater than that of feminine walk, while in feminine the hip movement is greater.

As mentioned earlier in section 4, the spinal cord can add noise to the action signals. The negative manipulation of the signals by the spinal cord is illustrated by Figure 13. The effect of the noise is visible on the left arm (pointed out by the red circle). This noise is caused by adding a random value to the data related to the elbow joint. As it is shown, the arm demonstrates some very rapid unexpected movements.

Figure 9. Addition of noise by the spinal cord

6. CONCLUSION

The aim of this research is to design and model a system in DEVS which can mimic the performance of the body in regards to producing and controlling human motion for both primary and secondary themes.

Various literature have modeled human motion. What lays in common in the discussed literature in section 2 is the fact that time discreteness or continuity as well as event discreteness or continuity is not discussed. The modeling approaches have thus far focused on describing the actual brain (responsible for synthesis and/or control of motion). While the effort for modeling the perfect brain for simulating human motion data can remain an ongoing research topic, rather different approaches towards modeling of the phenomena could result in development of new techniques for tackling the problem at hand.

In this research we have shown that DEVS are extremely accurate, straightforward, and efficient means for modeling human motion, therefore, discrete event A.I. systems and modeling approaches can be very promising for constructing of a comprehensive human motion synthesis and control system.

One other major advantage for this method of modeling the human motion synthesis system is the fact that different components of the model are stand-alone and can act independently. Thus, they can be represented by more complex coupled models and be enhanced in the future. For instance the brain in the proposed model can be later on manipulated to become more A.I.-based, or an entire model for the human body alone can be constructed and added to the current model for more accurate representation of the human body.

7. FUTURE WORK

To extend and improve the current work, more complex actions and sequences can be represented using this model. Also different learning techniques and even complex A.I.-based methods can be implemented in C++ and used in the brain component of the model. More emphasis on describing, controlling, and simulating secondary themes will also improve the system as secondary theme analysis is critical for synthesis of realistic motion data.

Real time DEVS are another possible field for further research. Virtual reality systems require a great deal of real time processing, thus are great candidates for employing a real time human motion data synthesis model.

References

[1] D. A. Forsyth, O. Arikan, L. Ikemoto, J. O'Brien, and D. Ramanan, Computational Studies of Human Motion: part 1, Tracking and Motion Synthesis, Foundation and trends in computer graphics and vision, Now Publishers, Volume 1, no 2/3, 2005.

[2] A. Hutchinson, Labanotation. Dance Books, 1996.

[3] G. A. Wainer, Discrete-Event Modeling and Simulation: A Practitioner's Approach. CRC Press, 2009.

[4] W. Park, D. B. Chaffin, B. J. Martin, and Y. Jonghwa, "Memory-Based Human Motion Simulation for Computer-Aided Ergonomic Design", IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, Volume 38, Issue 3, pp. 513 – 527, May 2008.

[5] J. Y. Zheng and S. Suezaki, "A model based approach in extracting and generating human motion", Proceedings of the Fourteenth International Conference on Pattern Recognition, Volume 2, pp. 1201 – 1205, Aug. 1998.

[6] R. Ahmad, M. S. Yaacob, and M. B. Che Omar, "Kinematics Modeling of Human Motion Using System Identification Technique", Second Asia International Conference on Modeling & Simulation, pp. 338 – 343, May 2008.

[7] W. Suleiman, A. Monin, and J. P. Laumond, "Synthesizing and Modeling Human Locomotion Using System Identification", 2006 IEEE/RSJ International

Conference on Intelligent Robots and Systems, pp. 1972 – 1977, Oct. 2006.

[8] R. Ahmad, M. B. Che Omar, M. S. Yaacob, M. Hussein, M. Zain, and M. Y. Abdullah, “Modeling of human motion through motion captured data”, 2008 International Symposium on Information Technology, Volume 3, pp. 1-7, Aug. 2008.

[9] C. Fanti, L. Zelnik-Manor, and P. Perona, “Hybrid models for human motion recognition”, 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Volume 1, pp. 1166 – 1173, June 2005.

[10] S. A. Etemad and A. Arya, “Recognition and Synthesis of 3D Human Motion with Personalized Variations”, Proceedings of 2009 IEEE International Conference on Multimedia Computing and Systems, pp. 106 – 111, April 2009.

[11] P. A. Stoll and J. Ohya, “Applications of HMM modeling to recognizing human gestures in image sequences for a man-machine interface”, Proceedings of the 4th IEEE International Workshop on Robot and Human Communication, pp. 129 – 134, July 1995.

[12] V. Pavlovic and J. M. Rehg, “Impact of dynamic model learning on classification of human motion”, Proceedings of the 2000 IEEE Conference on Computer Vision and Pattern Recognition, Volume 1, pp. 788 – 795, June 2000.

[13] S. A. Etemad and A. Arya, “3D Human Action Recognition and Style Transformations Using Resilient Back-propagation Neural Networks”, Proceedings of 2009 IEEE International Conference on Intelligent Computing and Intelligent Systems, pp. 296 – 301, Nov. 2009.

[14] N. Ogihara and N. Yamazaki, “Generation of human reaching movement using a recurrent neural network model”, Proceedings of the 1999 IEEE International Conference on Systems, Man, and Cybernetics, Volume 2, pp. 692 – 697, Oct. 1999.

[15] S. A. Etemad, P. Payeur, and A. Arya, “Automatic Temporal Location and Classification of Human Actions based on Optical Features”, Proceedings of the 2nd International IEEE Congress on Image and Signal Processing, pp. 1 – 5, Oct. 2009.

[16] Z. Tang, M. Joo Er, and C. J. Chien, “Analysis of human gait using an Inverted Pendulum Model”, 2008 IEEE International Conference on Fuzzy Systems, pp. 1174 – 1178, June 2008.

[17] S. A. Etemad, and A. Arya, “Modeling and Transformation of 3D Human Motion”, Submitted to 5th International Conference on Computer Graphics Theory and Applications.