Jurnal Teknologi

Full Paper

EDITING VIRTUAL HUMAN MOTION TECHNIQUES WITH DYNAMIC MOTION SIMULATOR AND CONTROLLER

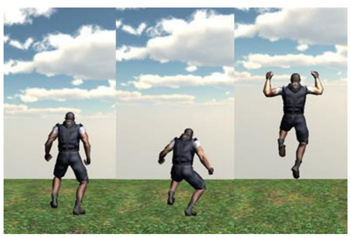
Ismahafezi Ismail, Mohd Sharizal Sunar*

Media & Games Innovation Centre of Excellence (MaGIC-X), Faculty of Computing, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia

Article history
Received
3 December 2013
Received in revised form
2 July 2014
Accepted
25 November 2014

*Corresponding author
shahrizal@utm.my

Graphical abstract



Abstract

Modifying realistic virtual human movement has become a challenging task to the researcher for computer games and animation development. To achieve realistic virtual human, the character movement must have same motion like real human. Virtual human movement can be created by blending different sources such as motion capture, dynamic and kinematics simulation. Editing dynamic movement requires a great skill from animator and takes a long time to setup. This paper presents a new technique for editing virtual human motion state using dynamic motion control in the real time animation. The system approach based on active dynamic control by normalizes the trajectory of vector space position. This technique explores the perfect balance in dynamic motion controls for virtual human motion initial and final states. For that purpose, an enhancement of proportional-derivative controller will be used. This paper focuses on three main parts; virtual human hierarchy, motion editing techniques and motion dynamic control.

Keywords: Virtual human, motion techniques, dynamic motion, simulator and controller

1.0 INTRODUCTION

Realistic and natural virtual human motion is very important part in computer animation industry. The virtual human motion in computer animation is not realistic and interactive as real human movement. To achieve realistic virtual character motion, a character should be adjusted so that a movement similar to humans in the real world. Analysis of actual human movement data can be applied to virtual human using keyframes or motion capture data. The main challenges in interactive computer games industry are to produce a dynamic character movement that can be manipulated as what animator wanted. This paper contributed a new technique to edit 3D character motion based on dynamic motion controller and simulator. From our system, new dynamic motion can be created from a normal movement while maintain the physical preferences details.

Motion controller and simulator is the main part of our system to manipulate the virtual human motion. Motion controller act like a system brain which processes the player input and checked all the initial setup. The motion controller main task is to calculate the acceleration of joint angles based on the latest situation of the insert motion data has been generated. After that, the simulator will update the current state data through a process of dynamic simulation. We can calculate the angular acceleration of output and difference of initial angular acceleration from this structure. Motion simulator task is to display all action needed. It processes the motion data from motion controller into world space motion. All the information data from character as frames, character rotation and transformation will be manipulated. The motion editor control approach based on active dynamic control by normalizes the trajectory of vector space position in real time animation. The example initial and final states for dynamic motion control is shown in Figure 1.

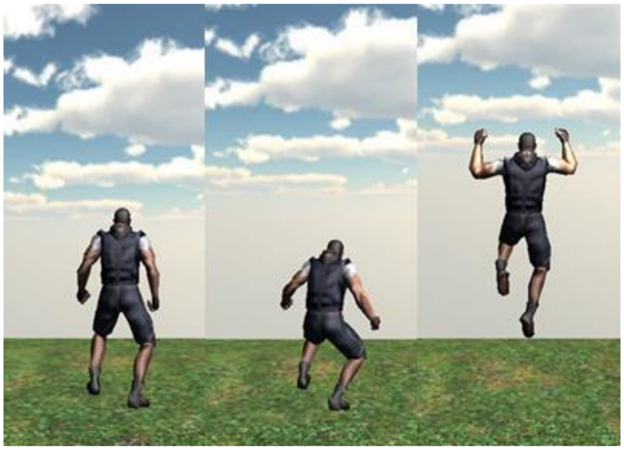


Figure 1 Example of dynamic motion control level from initial to final states in the system

In our system, animator can produce new motion without changing the keyframes or re-capture the animation using motion capture. Using a single motion data, user can manipulate interactively the dynamic motion as requested. User can manipulate added forces to get new motion with normal, dynamic or superhuman level. To maintain realistic motion, we introduce propotional-derivative control. The propotional-derivative control can balance the forces involve at the virtual human in dynamic motion level. Dynamic motion refers to the physical properties of 3D object, such as mass or inertia, and specifies how the external and internal forces interact with the object [1]. With the dynamic of character data, the control of the character's specific motion: walking, running, kicking, falling and jumping looks more realistic.

2.0 RELATED WORK

Several researchers have focused on the study of dynamic character motion. V.B. Zordan *et al*. [2] introduced a new method that allows characters to respond to unexpected changes in the environment based on the specific dynamic effects. The system generates a physics-based response and takes advantage of the realistic movement that is achieved by an actuated dynamic model of motion capture process. The dynamic character simulation responds to contact forces and determines the best plausible re-entry into motion library playback following the impact. To produce the results, a physically valid response will be created and the blending process will be generated into the desired transition-to motion. The dynamic motion controller will act in accordance with the upcoming motion. These techniques' main focus is to improve the character impact, but problems with responsive and interactive dynamic reaction for virtual human are still faced.

A. Shapiro *et al*. [3] focused on the "DANCE" platform for the development of physically based controllers for articulated figures. The main aim of this platform is to train an inexperienced user to develop dynamic controllers. In 2007, the system has been improved by creating a toolkit for dynamic articulated characters controllers [4] under the physical simulation. The dynamic character controllers developed by using key-framed based control, reduced dimensionality physics, scripting controllers via a controller language, and interactive control of dynamic characters. However, this technique cannot perform a complete motion stage while interacting with the environment.

"Dynamo" is a technique that allows a character to set and maintain poses robust to dynamic interactions introduced by P. Wrotek *et al*. [5] in 2006. The system produces physically plausible transitions between motions without directly using a blending process. The main idea is to apply torques to match the desired world-space pose and maintain root orientation. After that, the motion blending emerges from continual simulation. The main weakness of this system is that it cannot collaborate with implausible situation such as displaying super-human abilities.

Y. Abe and J. Popovic [6] worked on a control algorithm that generates realistic animations by incorporating motion data into task execution. The system's focus is on interactive animation of dynamic manipulation tasks such as lifting, catching, and throwing. This interactive system allows Cartesian space force limits. The method always provides new command vectors that produce manipulation. This control algorithm has problems with the loss of control over some degrees of freedom. The motion stage will not be completed without pre-planning of the torques of motion.

Allen [7] is another researcher who worked on virtual human dynamic motion controller. His approach focuses on timing constraints using a natural looking motion and allows a realistic response. However, the algorithm does not take into account subsequent effects for Parent-Childs concept using torques.

U. Muico *et al*. [8] proposed a nonlinear control system using character contacts to emulate motion-capture data. The framework uses nonlinear controllers with a large set of different styles of possible motions. The drawback of this control system is it cannot recover from larger changes in the environment because that requires intentional deviation from pre computed reference trajectories.

Recently, Kenwright *et al*. [9] proposed a real-time modeling of 3D skeletal motion with balancing properties. They described an approach in modeling mid-to-lower body of 3D human movement in real-time. The dynamic motion in this research did not cover upper part of the body and human behavior interaction. Another latest research on 3D Character motion can be found in [10]. They concentrated on two strategies in dynamic motion control: non-linear time scaling of joint trajectories and how to modify the joint angles directly.

3.0 VIRTUAL HUMAN HIERARCHY

Generally, virtual human motion development in computer animation involves joints of 3D models controlled by the skeleton hierarchy. These joints have been combined with three-dimensional geometric models, such as polygonal mesh. Animator try to make complex character movement by control all this joints parameter.

Currently, motion capture techniques have been used widely from video game animation to computer graphic effects in movies. An actor, are placed in a special suit containing sensors that record the motion to get the real human movement data. The motion data output is still not perfect because need to clean up from keyframe animator to make it look more natural [11]. However if we compare to other techniques, the output data from motion capture technology shows more realistic, convincing and better character movement than other techniques.

3D character movement controlled using skeleton structure or hierarchy. A hierarchy uses grouping or parenting concept. For example, of human leg, the hip is the parent of the upper leg. Meanwhile, the lower leg is the child the upper leg, and the foot is the child of the lower leg. 3D character hierarchical model have smaller number of parameter that give consistency to dynamic motion. In real time animation environment, each bone depends on the orientation and the joint with its parent. 3D Character motion, *m(t)* can be derive as:

$$m(t) = (p_r(t), q_r(t), q_1(t) \cdots, q_n(t)) \qquad (1)$$

Where pr(t) and qr(t) represent the position and the orientation of the root bone. The orientations of the rest of the n bone in hierarchy are q1(t) ....qn(t) refer to the coordinate systems of their parent at time. Generated motion has to enforce with physical law of motion for creating the realistic virtual human motion [12].

Figure 2 shows parent body and required translation to the child body for virtual human. After transform the child body into global space, we can calculate the new position of the body. The parent body rotation matrix defined as *R*0 and parent body position as $\vec{p}_0$ When the parent body move, automatically the child body transform from local body space to global space. The parent body position transformation:

$$\vec{p}'_o = R_0\vec{l} + \vec{p}_o \qquad (2)$$

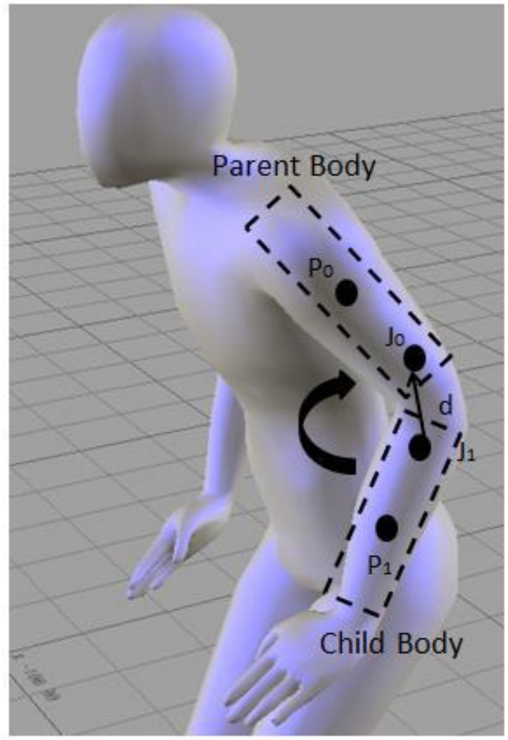


Figure 2 Virtual human parent-child body relationship

Meanwhile, the child body is defined as R1 and $\vec{p}_0$. When the parent body moves, the joint J0 and J1 change to the same coordinate place. To tranform child body from local space into global space:

$$\vec{j}'_n = R_n\vec{j}_n + \vec{p}_n \ \text{, n=0,1} \qquad (3)$$

To get the vector $\vec{\vec{d}}$, we substract $\vec{j}_0$ and $\vec{j}_1$. the new body position, $\overrightarrow{p'}_1$:

$$\vec{p}'_1 = \vec{p}_o + R_o\vec{j}_o - \vec{R}_1 j_1 \qquad (4)$$

The function of inverse kinematics and forward kinematics is to calculate the bone's position, including the joints position and angles [13]. Normally, inverse kinematics is used for motions involving the lower part of a character's body such as joints from the foot to the pelvis [14, 15, 18]. It is very hard to use forward kinematics because the body position will be moving below the surface or ground. This transaction makes the character's motion very unreliable and unconvincing.

In the case of shoulder rotation, or to get the position of the upper arm, we use forward kinematics calculation [17]. Forward kinematics is a top-down technique rotation used to position the character's upper body part in real time animation. Each skeleton joint has its local transformation, and parent transformation will determine the global transformation of each skeleton joint.

4.0 MOTION EDITING TECHNIQUES

Virtual human realistic motion is becoming more high demand in the video game and film industries. Its importance is beyond any doubt being a key aspect in films, where most of them have a large part of the movie based on computer animation, and in the video game industry, where real-time animations are becoming more and more realistic [20], improving the user experience. Virtual human has a rigid body that has its own forces, velocity, mass and physical properties. The main step to control dynamic motion simulation is the structure to balance the virtual human movement.

Although physic properties can be applied to character motions in real time animation, however it is still limited when it comes to rigid objects. Our system focused on added forces to the character motion while proportional-derivative controls maintain the trajectory balance in the vertical axis of motion. The main structure of our system is shown in Figure 3.

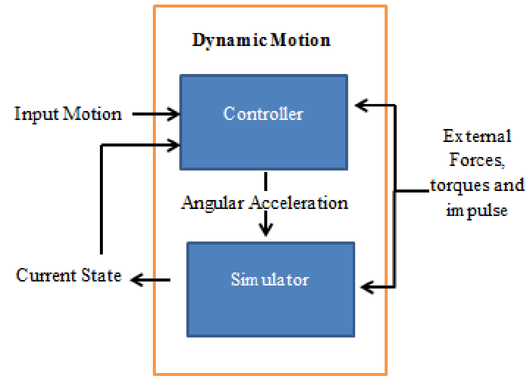


Figure 3 Dynamic motion editor with controller and processor as the main structure

A Motion Controller and Simulator

In general, dynamic motion editor must have two core parts: controller and simulator. The structure of our system motion is shown in Figure 3. Using controller function, we can calculate the angular joint acceleration directly by referring to the latest state of the motion capture data input. After that, the simulators update the process through dynamic character motion. From Euler angles Equation [16], the orientation of the body frame is:

$$\theta = \theta_x i + \theta_y j + \theta_z k \tag{5}$$

Where Θ_x, Θ_y and Θ_z are scalars, and i, j and k are the world coordinate axes. From this equation, we calculate the angular velocity:

$$\omega = \frac{d\theta_x}{dt} i + \frac{d\theta_y}{dt} j + \frac{d\theta_z}{dt} k \tag{6}$$

The previous result from this structure is the output angular acceleration as the sum of $\ddot{\Theta}_{initial}$ and the difference of the angular acceleration $\Delta\ddot{\Theta}$. Meanwhile, the results input based on combination of a human body model and external physical input for the controller and the simulator. Our main structure combined the active control torque and other external physical interaction. The output motions have been generated by the physical simulator. Using this system, user can control the basic dynamic simulation of character movement. The methods linking the connection between forces acting to the body and acceleration can be categorized in two classes: maximal coordinate and reduced coordinate.

B An Enhancement of Proportional-derivative Controller

A lot of approaches have been developed for the purpose of character motion control based on dynamic [22]. Our new method can calculate and change the motion speed and trajectory using the proportional-derivative control. Our technique is creating motion using inverse kinematics method and produces a human walking motion using inverse dynamic.

3D character movement described by its mass, m, and its trajectory, r(t) [19, 21]. We need to get positive gradient for increasing character velocity as show in Figure 5. Giving the time constraint in our calculation, the character velocity is:

$$v(t) = \lim \Delta t \to 0 \frac{r(t+\Delta t) - r(t)}{\Delta t} = \dot{r}(t) \tag{7}$$

So, the 3D character acceleration is defined as the limit change of velocity:

$$a(t) = \lim_{\Delta t} \to 0 \frac{v(t+\Delta t) - v(t)}{\Delta t} = \dot{v}(t) \tag{8}$$

We derive linear momentum, P, of 3D character defines using Newton's second law equations with forces, F has both magnitude and direction:

$$\mathbf{P} = m\mathbf{v} \tag{9}$$

$$F(t) = \frac{dP(t)}{d(t)} = m\frac{dv(t)}{d(t)} = ma(t) \tag{10}$$

Using linear momentum, we calculate total external forces involves in character action:

$$\sum_i F_i(t) = m \sum_i a_i(t) \tag{11}$$

5.0 MOTION DYNAMIC CONTROL

Dynamic motion need to be realistic and natural in computer animation. Figure 4 shows that our motion editor has a vertical trajectory control based on proportional-derivative method. The system can edit dynamic action such as jumping, front flip, running and walking. We can manipulate the original virtual human motion and change it to look more dynamic such as superhuman motion or less dynamic such as weaker movement than natural movement.

Figure 4 Motion control level interface in motion editor system

A Basic Movement: Running and Walking

We control the forward and backward speed by normalize the vector position to horizontal axis. The virtual human directions need to transform to world space relative to character orientation. Using speed level limiter, user can manipulate normal state motion of walking to fast walk or slow walk.

B Front Aerial Flip

We control the acrobatic speed with increasing the velocity direction to vertical axis as show in Figure 5. If we constraints time with increased position, we can get faster flip action refer to the root orientation and the maximum height of flip increasing. We can also make the action less dynamic by the limitation of the flip speed after character leave the ground. This system also has mass manipulation by using different character mass with a same action speed. If we reduce the character mass, we can get more dynamic action. The reactions from front flip motion show in Figure 7(a).

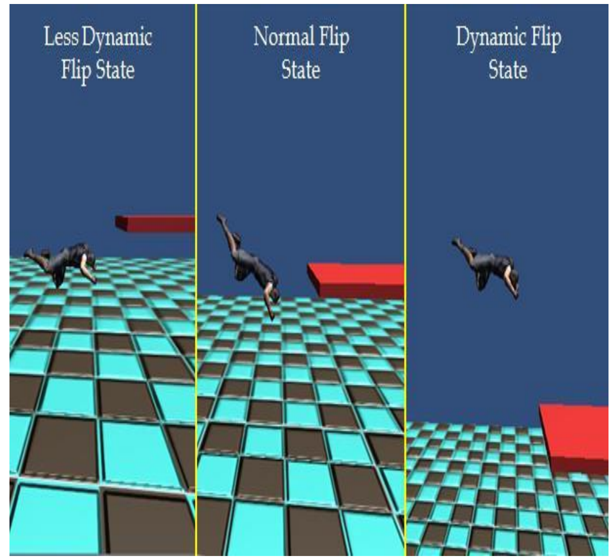


Figure 5 Edited front flip action

C Level of Jumping

We edit jumping motion by control root joint orientation and angular velocity around the vertical axis. The interactive editor shows in Figure 6 for virtual human jumping process. We can manipulate the virtual human original jump motion from normal human jump to superhuman jump. This interactive editor can help animator to find the natural dynamic jumping that suitable with their 3D character. The interaction between jump motion and external forces make the character movement look like a believable action. For example, the levels of jumping motion show in Figure 7(b).

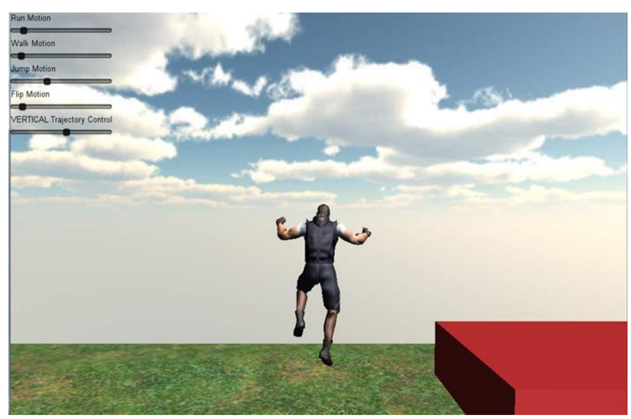
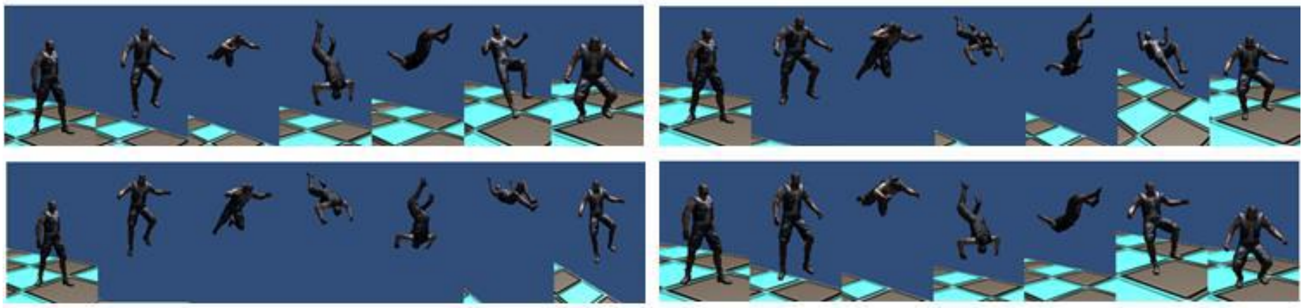
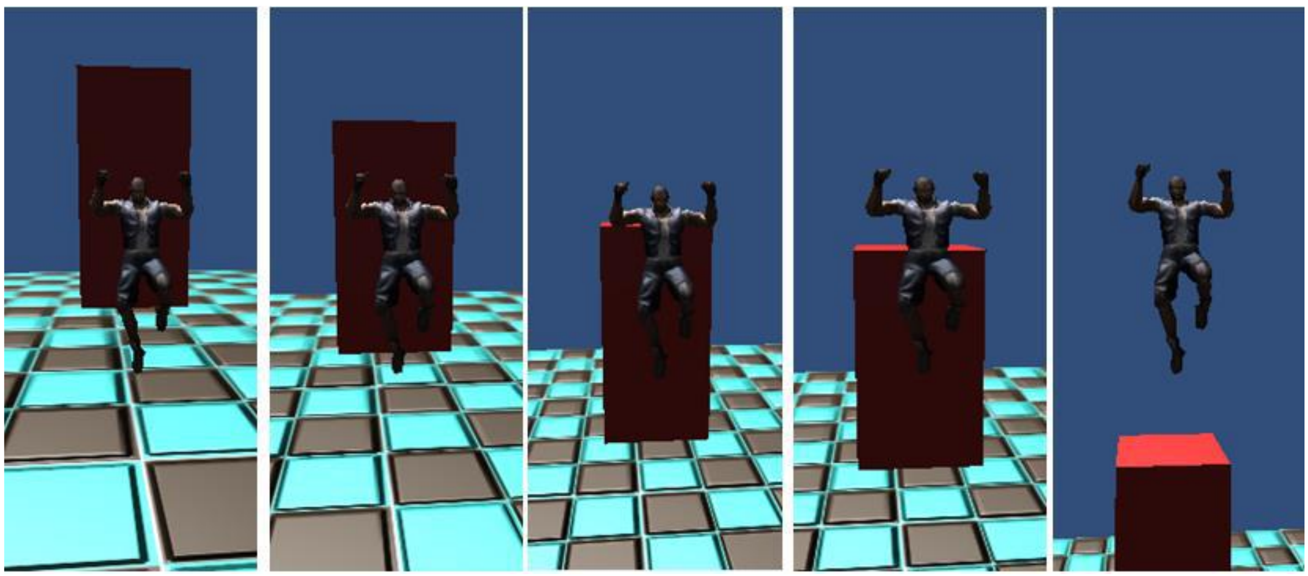


Figure 6 Interactive motion control for jumping state



(a) Frontal Flip. (Left Top) The motion edited with decreasing additional force. (Right Top) Original frontal flip motion . (Left Bottom) The motion edited with increasing additional force. (Right Bottom) The motion edited with additional mass and force



(b) Jumping State. Edited to make a lower and higher jump. The original motion shows in the middle image.

Figure 7 Experimental results

6.0 CONCLUSION

In this paper, we present new techniques for editing virtual human motion with trajectory control and proportional-derivative control. Our system can calculate and change the motion speed and trajectory using the joints of skeleton. Our approach involves two main parts: motion controller as a brain and motion simulator as a motion processor. The system created show that it is possible to manipulate 3D character motion and make it more interactive and dynamic. Our system can manipulate the dynamic action but not a long sequence virtual human movement. Exploring the perfect balance between motion editor for virtual human and object interaction still cannot achieve. The main challenge for character motion in real time animation is to make the character move automatically and instructed like real human.

Nowadays, researchers are trying to find a perfect balance between motion control and sophisticated long sequence interaction. Researchers need to control large data sets and automatic methods for mapping the correct input data into local models. Multiple learned models and different control methods need to be explored for the purpose of getting a natural, balanced dynamic character motion while maintaining the character's physical properties.

Acknowledgement

The research paper supported by UTM-IRDA Digital Media Centre, MaGIC-X (Media and Game Innovation Centre of Excellence), Universiti Teknologi Malaysia using Exploratory Research Grant Scheme (ERGS) vot number R.J130000.7828.4L092. Special thanks to Ministry of Higher Education (MOHE) and Research Management Centre (RMC) providing financial support of this research.

References

[1] Hu, Y., S. Wu, S. Xia, J. Fu and W. Chen. 2010. Motion Track: Visualizing Variations of Human Motion Data. *Pacific*

Visualization Symposium (PacificVis). Taipei, Taiwan. 2-5 March 2010. 153-160.

[2] Zordan, V.B., A. Majkowska, B. Chiu and M. Fast. 2005. Dynamic Response for Motion Capture Animation. *ACM Transactions on Graphics (TOG)*. 24(3): 697-701.

[3] Shapiro, A., P. Faloutsos and V.N.T. Hing. 2005. Dynamic Animation and Control Environment. In *Proceedings of Graphics Interface*. Victoria, British Columbia. 9-11 May 2005. 61-70.

[4] Saphiro, A., D. Chu, B. Allen and P. Faloutsos. 2007. A Dynamic Controller Toolkit. In *Proceedings of the ACM SIGGRAPH Symposium on Video Games*. San Diego, CA, USA. 4-5 Aug. 2007. 15-20.

[5] Wrotek P., O.C. Jenkins and M. McGuire. 2006. Dynamo: Dynamic, Data-Driven Character Control with Adjustable Balance. Proceedings of the 2006 ACM SIGGRAPH Symposium on Video Games. Boston, MA, USA. 29-30 July 2006. 61-70.

[6] Abe, Y. and J. Popović. 2006. Interactive Animation of Dynamic Manipulation. In *Eurographics Symposium on Computer Animation*. Vienna, Austria. 2-4 Sept. 2006. 195-204.

[7] Allen, B., D. Chu, A. Saphiro and P. Faloutsos. 2007. On the Beat!: Timing and Tension for Dynamic Characters. In Eurographics Symposium on Computer Animation. San Diego, CA, USA. 3-4 Aug. 2007. 239-247.

[8] Muico, U., Y. Lee, J. Popović and Z. Popović. 2009. Contact-Aware Nonlinear Control of Dynamic Characters. *ACM Transactions on Graphics (TOG)*. 28(3): 1-9.

[9] Kenwright, B., R. Davison and G. Morgan. 2011. Dynamic Balancing and Walking for Real-Time 3d Characters. *Proceedings of the 4th International Conference on Motion in Games*. Edinburgh, UK. 13-15 November 2011. 63-73.

[10] Munirathinam, K., S. Sakka and C. Chevallereau. 2012. Dynamic Motion Imitation of Two Articulated Systems using Nonlinear Time Scaling of Joint Trajectories. In *International Conference of Intelligent Robots and Systems (IROS)*. Vilamoura. 7-12 October 2012. 3700-3705.

[11] Oshita, M. 2006. Motion-Capture-based Avatar Control Framework in Third-Person View Virtual Environments. *Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*. Hollywood, CA, USA. June 14-16, 2006. 1-9.

[12] Cavazza, M., R. Earnshaw, N.M. Thalmann and D. Thalmann. 1998. Motion Control of Virtual Humans. *IEEE Computer Graphics and Applications Archive*. 18(5): 24-31.

[13] Moeslund, T. B., A. Hilton and V. Krüger. 2006. A Survey of Advances in Vision-based Human Motion Capture and Analysis. *Computer Vision and Image Understanding*. 104(2): 90-126.

[14] Shilei, L., L. Jiahong, B. Wu, L. Chen, Z. Hu, J. Su & Y. Bai. 2009. A Survey of Dynamic Motion Generation and Control for Virtual Characters. *International Conference on Computational Intelligence and Software Engineering*. Wuhan. 11-13 December 2009. 1-5.

[15] Ismail, I., H. Kolivand, M. S. Sunar and A. H. Basori. 2013. An Overview on Dynamic 3D Character Motion Techniques in Virtual Environments. *Life Science Journal*. 10(4): 1-9.

[16] Popovic, Z. and A. Witkin. 1999. Physically based Motion Transformation. *Proceeding of the 26th annual conference on Computer graphics and interactive techniques (SIGGRAPH 99)*. LA, California, USA. 8-13 August 1999. 11-20.

[17] Geijtenbeek, T. and N. Pronost. 2012. Interactive Character Animation using Simulated Physics: A State-of-the-Art Review. *Computer Graphics Forum*. 31(8): 2492–2515.

[18] Safona, A., J. K. Hodghins, N. S. Pollard. 2004. Synthesizing Physically Realistic Human Motion in Low Dimensional, Behaviour- Specific Spaces. *Journal of ACM Transactions on Graphics (TOG)–Proceedings of ACM SIGGRAPH 2004*. 23(3): 514-521.

[19] Szczuko, P., B. Kostek, & A. Czyzewski. 2009. New Method for Personalization of Avatar Animation. In *K. A. Cyran et al. (Eds.): Man-Machine Interactions*. 435-443.

[20] Woong, C., T. Ono, & K. Hachimura. 2009. Body Motion Analysis for Similarity Retrieval of Motion Data and Its Evaluation. *Intelligent Information Hiding and Multimedia Signal Processing Conference*. Kyoto, Japan. 12-14 September 2009. 1177-1180.

[21] Ho, E. S. L., T. Komura, & C. L. Tai. 2010. Spatial Relationship Preserving Character Motion Adaptation. *ACM Trans. Graph*. 29(4): 1-8.

[22] Ismail, I., M. S. Sunar, M. K. M. Sidik, & C. S. Yusof. 2011. A Review of Dynamic Motion Control Considering Physics for Real Time Animation Character. *Workshop on Digital Media and Digital Content Management*. Hangzhou, Zhejiang, China. 15-16 May 2011. 86-90.