Immersive Simulation and Training of Person-to-3D Character Dance in Real-Time

Salva Kirakosian
Electrical and Computer Engineering
Technical University of Crete
subkiro@gmail.com

Emmanuel Maravelakis
Electronic Engineering
Hellenic Mediterranean University
marvel@chania.teicrete.gr

Katerina Mania
Electrical and Computer Engineering
Technical University of Crete
k.mania@ced.tuc.gr

***Abstract*—This paper presents a person-to-3D character, immersive, real-time dance training framework for leaders. Rather than immersive mimicking of pre-recorded dance animations as in previous work, the user is now trained to lead a 3D partner while dancing, initiating movements to which the 3D character responds to, in real-time. Realistic animations were applied to the rigged 3D human model of a dance partner using Inverse Kinematics (IK). The 3D dance partner is rigged based on custom skeletons using 3D geometrical bones and IK solvers. Motion capture data were applied to the rigged 3D model to create realistic dance animations. Initially, the application features a tutorial guide process. The 3D character is standing facing the user who selects dance training options using gestures such as thumb and waving hands, touches and pinch actions. The 3D character partner responds in real-time, producing body motion according to the users' dance leading actions following Salsa dance rules. The experience is evaluated as realistic dance. The software architecture could be utilized in a wide range of training systems in diverse domains, such as training in the manufacturing domain.**

Keywords*—*Immersive Simulation and Training, Serious Games, Inverse Kinematics, Motion Capture

I. Introduction

Immersive training could be employed for a wide range of applications for entertainment, education or simulation. For instance, in the manufacturing domain, training is expensive and also dangerous. The software infrastructure for immersive software development has similarities across domains. It includes scenes, which could be imported into the system, processes, interactions and animation, involving 3D characters or not, based on the application. Although this paper is focused on entertainment, the software infrastructure developed could be applied to other domains too [1]. Latin dance is a social activity which requires the physical presence of two people interacting. Previous dance training approaches utilized motion capture technology to capture dance routines and, then, reproduce them. Often, VR dance or sports' training systems merely replay captured dance sequences through Head Mounted Displays (HMDs) in order for the user to learn by mimicking displayed movements without providing any feedback [2], [3]. Later efforts captured a dance trainee's motion against the desired (teacher's) dance steps and applied this motion onto a virtual character. Students participating in this process would receive feedback on the accuracy of their performance, however, there was no capability of actual live dancing with a 3D character acting as a partner [4], [5], [6]. Recent research presented a system for the real-time assessment and visualization of ballet dance movements, as performed by a student in an instructional VR setting. After capturing skeletal joint tracking, the performance evaluation was conducted in a 3D CAVE. Although the CAVE allows wide-area viewing of the virtual teacher, dance recognition performance and feature extraction is complex [7], [8]. The fact that there was no interactivity with the 3D expert limits the dance training effectiveness [9]. Other work has tracked user's dance gestures by visualizing matching dance routines, however, again, there was no physical or virtual association of the learner and teacher [10]. Recent research presented a non-technical conceptual model of utilizing touch, vision and hearing allowing multiple users to dance together, without having to be physically present in the same space using Augmented Reality and holography [11]. A process based on Hidden Markov Models (HMM) for learning the structure of the dance motions of a leader and follower predicts the intention for the following movement [12], [13]. A unified framework for interactively synthesizing movements by virtual partners in response to the user-controlled character has been proposed [14]. The system infers the intention of the user by selecting one of pre-recorded interactions to facilitate motion synthesis. Such methods, as above, are not integrated in a dance training framework. Recent work puts forward a serious game for folklore Greek dance training as well as dance archiving. Joints' movements are extracted using the Labanotation system [15]. While this work is valuable for recording, archiving and demonstrating folklore dancing, it does not offer interactive dance training.

In this paper, we propose a person-to-3D character immersive real-time dance training application for leaders. The user can interact with a virtual character using a low-cost, easy-to-use system which includes a Head Mounted Display and a hand-tracking device (Figure 1). The user is trained in Latin dance by either a 3D humanoid character or a 3D scanned human dancer who responds in real-time to the users' (leader) improvised dancing. The leader asks for the hand of the 3D character (follower) as in real life. Realistic interaction of the 3D character with the human dancer were applied using Inverse Kinematics (IK). The 3D character is rigged based on custom skeletons and 3D geometrical bones as well as IK solvers. Motion capture data are applied to the rigged 3D model to create realistic dance animations. Captured animations are imported to an implemented Finite-State Machine (FSM),controlling the animation transitions. The training framework presented was designed so as to be expanded to include additional movements and diverse training material (Figure 2).

Fig. 1. Set-up

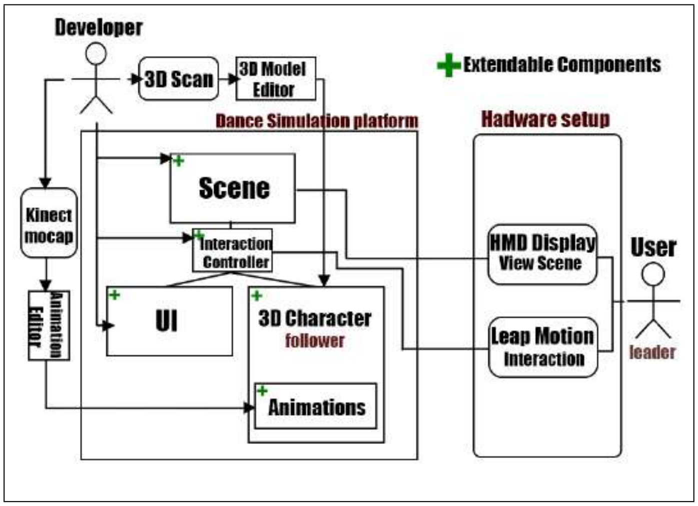

Fig. 2. System Architecture

II. Implementation

A. Character Modeling and User Interface

The dance trainee can select one of two 3D character options, e.g., a humanoid 3D character or a 3D scanned model of a human dancer (Figures 3, 7). The professional Spanish Latin dancer **Alberto Rodriguez** was invited and scanned using the Artec Eva structured light 3D scanner by Artec3D. The Artec Eva uses ”White scanning” to generate the 3D model of an object. White Scanning consists of gathering surface height measurements of an object using ”white light” illumination. The method adopted for scanning so that the final 3D model was produced, consisted of four steps (Figure 3): **Scanning a physical person:** The dancer was standing on a T pose while the 3D scanner was moved around the dancer, held by a human operator, scanning all parts of his body. **Repairing 3D scanned geometry:** Slight movement of the human model during the 3D scanning procedure caused certain geometrical inaccuracies. For this reason, a separate scanning of specific body parts was conducted so that the initially failed scanned parts were repaired. A manual repair and connection of these parts was conducted. **Geometry optimization:** After the 3D scanning was completed and the body parts connected, the resulting 3D model included more than 1 million vertices. The resolution (amount of vertices) was reduced to 90% lower than the original scanned model, retaining face features in higher detail so that the dance trainee could intuitively perceive that the 3D character represents a realistic human model. **Biped Rigging:** Finally, bones to the skin were added so that animations could be applied.

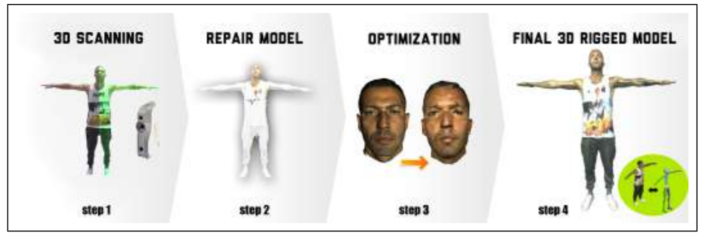

Fig. 3. 3D scanning of the Spanish Latin dancer Alberto Rodriguez

Fig. 4. User Interface

A simple and intuitive User Interface (UI) was developed for inexperienced dancers. Each selection included in the UI canvas represents a button that the users can touch. The user’s hands are tracked and these movements are reflected onto the 3D hand model visualized for interaction (Figure 4). Initially, the 3D character is standing just below the main UI facing the user, offering instructions in relation to the dance training functionality. The users practice their actions by interacting with UI panels using gestures such as thumb and waving hands, touches and pinch actions. Experienced VR users can follow the main dance tutorial, selecting one out of three basic options. The dance training process starts with the first option named ”Basic Steps” of the Salsa Latin dance which includes the ”Basic A”, the ”Side Steps” and the ”Right Turn” dance routine. The user learns to execute the basic steps that the 3D character demonstrates and not the leading movements. By selecting the ”Interaction” option, the users can practice leading abilities using their hands leading naturally the 3D model to execute the ”Side Steps” and the ”Turn” animation while both execute the ”Basic step”. Finally, the users can select the last ”Dance” option, where they can improvise and dance, leading the 3D character to basic ”Side Steps” and ”Turn”. The 3D character (follower) responds in real-time to the users’ dance leading actions. Importantly, the users’ hand gestures are tracked while joined with the 3D character’s hands. The improvised dance starts when the user’s and the 3D character’s hands are virtually connected. The main application scene is populated by two characters. The first is the dance trainee and the second character is a 3D character the user interacts with. The users can view their hands through the HMD represented by 3D hand models (Figure 7, 8). The hand movement and positions are calculated in real-time using the

Leap Motion sensor as shown in Figure 1. The user can interact with the UI elements and virtual character by extended gestures and virtual touches, or even through basic gestures such as pinching faraway buttons.

B. Character Animation and Inverse Kinematics

In order to animate a 3D humanoid character, this should be rigged as shown in (Figure 3) step 4. As a first step, we produced pre-rendered animations representing specific states such as "Idle state", "Flexing state", "Walk state", "Turn state". We execute these animations as the core of the Finite State Machine as shown in Figure 6. Human motions were captured with a tracking device. Captured data was transferred onto the 3D humanoid model (follower) producing realistic dance movement animations. The captured data was not directly connected to the main 3D model but to the generic skeleton while the record mode was enabled through the *Motion Builder platform* combined with Microsoft Kinect v1. After the recording is finished, the animations were optimized by smoothing the jitters. Finally, the animation was ready to be applied to the humanoid rigged 3D characters in Unity. Real-time tracking of the user's hands was possible based on a standard motion tracking device (Leap Motion). No tracking data is stored in the system. In order for a user to smoothly interact with a humanoid 3D character while holding its virtual hands, Unity's Inverse Kinematics (IK) system was customized to execute dynamic animation of the 3D character. Forward and Inverse Kinematics are similar to a function and its inverse. In robotics, for example, kinematics refers to calculating the relations between end-effectors and joint angles. In relation to Forward Kinematics (FK), the joint angles are the input and the coordinates of the end-effectors are the output. In relation to Inverse Kinematics (IK), the given input is the set of coordinates of the end-effectors and the output to be calculated is the set of joint angles. For a multiple Degrees-of-Freedom (DoF) robot, calculating the FK is quite straightforward. However, IK calculation could be tricky in relation to the same end-effectors coordinates, lacking a unique configuration (or solution), especially when the system is redundant. In order to use Inverse Kinematics (IK) for the 3D character we need to create a reference to the users Left and Right hands: Right Hand IK and Left Hand IK. In order to implement the interaction between the 3D humanoid character and the user's 3D hand model, a GameObject was created in Unity named **LeftIKControl and RightIKControl** and added to the Left and Right 3D hands model of the user respectively. These IK Controllers contain a child GameObject named **Lsphere & Rsphere**. Lsphere and Rsphere include a collider. A rigidbody component is used as a reference in IK of the Left and Right hand of a 3D character. When the GameObjects with tag "LeftSphere and RightSphere" enter the 3D character's zone of interaction, the sphere collider is used as a trigger as shown in (Figure 5) and a function is activated so that the Right IK and Left IK are set to Enable. By using the position of the user's hands while they are inside the interaction zone, we can alter the predefined animations in regards to the user's "leading" of the "partner's" hands using IK.

C. Finite State Machine (FSM) and AI

A 3D character animator controller based on a State Machine was developed to be applied to any humanoid character in the scene. Animation layers were defined for managing complex state machines for different body parts of a 3D character (Figure 6). For example, in a simple waving animation of a 3D character, a lower-body layer represents the idle state and an upper-body layer represents the waving hand state. This technique allows the 3D character to wave a hand while in idle mode. The Artificial Intelligence (AI) navigation system was customized to allow the characters to move around the game world following the user, using navigation meshes that are created automatically based on the scene geometry. The navigation of the characters can, then, be adjusted at run-time. The distance the user needs to reach the 3D humanoid character is minimized, by guiding the 3D character to move towards the user. The stop distance was defined to 4 units in order to enable the user to walk to the interaction zone taking a minimum amount of steps (Figure 5).

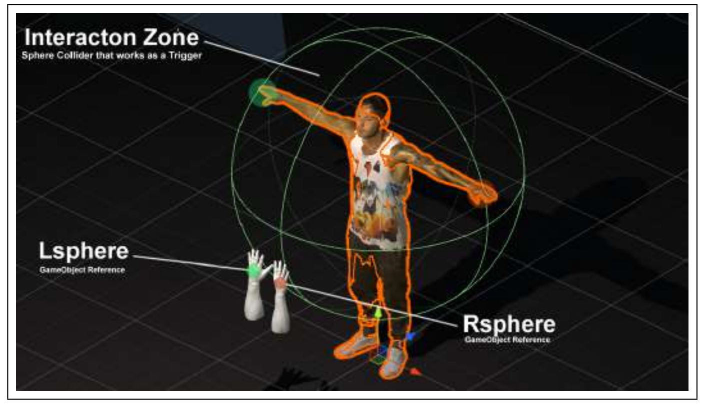

Fig. 5. Interaction Zone

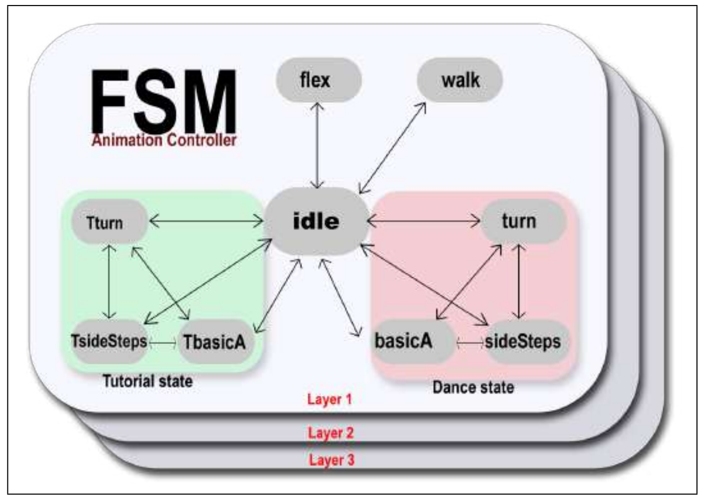

Fig. 6. Finite State Machine (FSM) & Layers

III. Evaluation and Conclusions

Besides pilot studies which largely improved interaction throughout development, 20 adults aged 18-40 were invited for system evaluation. The main instrument for evaluation was the Immersive Experience Questionnaire (IEQ) consisting of questions in 6 categories, e.g., attention, temporal disassociation, temporal transportation, challenge, emotional attachment and enjoyment [16]. Moreover, the standard Simulator Sickness

Fig. 7. Dance Training with Humanoid

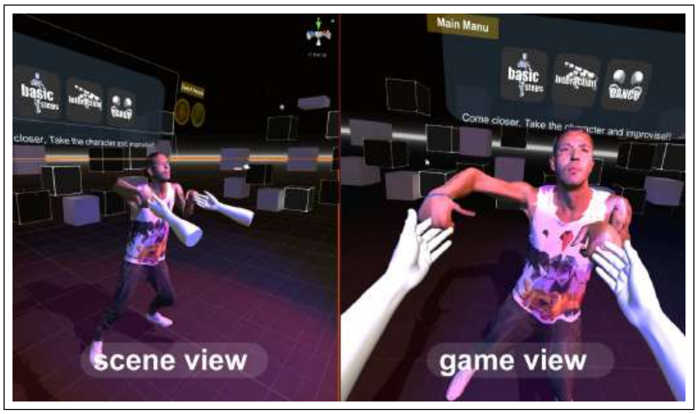

Fig. 8. Dance Training with 3D Human Character (scanned)

Questionnaire (SSQ) was also administered. The questions were rated from [min = 1 to Max =5], with 1 signifying the minimum grade and 5 the maximum. The IEQ results obtained from the participants were all by far above average. The highest values were associated with attention (mean 4.37), emotional attachment (4.46), enjoyment (4.40) and immersion (3.87). The SSQ indicated minor motion sickness symptoms. Overall, users were impressed by the capabilities of the VR dance training system. Almost all participants (18=positive,2=neutral, 0=negative) felt more confident to invite someone to dance in real life after they engaged with VR dance training.

We demonstrated that it is possible to provide a fully interactive, immersive training system, with minimal to non-perceptible latency, advancing current dance training systems which mostly demonstrate pre-rendered dance animations the trainees are asked to imitate. The user is able to lead a 3D character in Latin dance by joining real and virtual hands and the 3D character can respond, through IK solvers, to users dance actions in real-time and in a non-predetermined manner (Figures 7,8). Future improvements may include enhancing the photorealism of the scene by adding real-time shadows and lighting. It would be desirable to analyze music clips added in the game, so that the tempo and the beat of the soundtrack are tracked. Then, animations can be synchronized to the tempo of the music. The 3D humanoid character is, then, going to be able to execute movements based on the music tempo. Animations could be more complex, depending on the capturing device and method so that the actions executed and dance movement combinations are enriched. The software infrastructure could also be modified for training in diverse domains such as in simulation, manufacturing, virtual shops' training etc.

Acknowledgment: This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH – CREATE – INNOVATE (project code:T1EDK-01469).

REFERENCES

[1] M. Frieder F. Loch, Fahimipirehgalin, J. N. Czerniak, A. Mertens, V. Villani, L. Sabattini, C. Fantuzzi, and B. Vogel-Heuser. An adaptive virtual training system based on universal design. *IFAC-PapersOnLine*, 51(34):335–340, 2019.

[2] Ungyeon Yang and Jeonghyun Kim. Implementation and evaluation of "just follow me": An immersive, vr-based, motion-training system. *Presence: Teleoperators and Virtual Environments*, 11(3):304–323.

[3] K. Hachimura, H. Kato, and H. Tamura. A prototype dance training support system with motion capture and mixed reality technologies. In *RO-MAN 2004. 13th IEEE International Workshop on Robot and Human Interactive Communication*, pages 217–222.

[4] D. Alexiadis, P. Kelly, P. Daras, N. O'Connor, T. Boubekeur, and M. B. Moussa. Evaluating a dancer's performance using kinect-based skeleton tracking. In *Proceedings of the 19th ACM International Conference on Multimedia*, MM '11, pages 659–662, 2011.

[5] J. C. P. Chan, H. Leung, J. K. T. Tang, and T. Komura. A virtual reality dance training system using motion capture technology. *IEEE Transactions on Learning Technologies*, 4(2):187–195, April 2011.

[6] M. Raptis, D. Kirovski, and H. Hoppe. Real-time classification of dance gestures from skeleton animation. In *Proc. 2011 ACM SIGGRAPH/Eurographics Symp. on Computer Animation*, pages 147–156.

[7] L. Deng, H. Leung, N. Gu, and Y. Yang. Real-time mocap dance recognition for an interactive dancing game. *Computer Animation and Virtual Worlds*, 22(2-3):229–237, 2011.

[8] Y. Yang, H. Leung, L. Yue, and L. Deng. Automatic dance lesson generation. *IEEE Trans. Learn. Technol.*, 5(3):191–198, January 2012.

[9] M. Kyan, G. Sun, H. Li, L. Zhong, P. Muneesawang, Nan N. Dong, B. Elder, and L. Guan. An approach to ballet dance training through ms kinect and visualization in a cave virtual reality environment. *ACM Trans. Intell. Syst. Technol.*, 6(2):23:1–23:37, 2015.

[10] J. Tang, Jacky J. C. P. Chan, and H. Leung. Interactive dancing game with real-time recognition of continuous dance moves from 3d human motion capture. In *Proc. of 5th ACM Int. Conf. on Ubiquitous Information Management and Communication*, pages 50:1–50:9.

[11] V. Saxena, T. Feldt, and M. Goel. Augmented telepresence as a tool for immersive simulated dancing in experience and learning. In *Proc. of ACM India HCI 2014 Conference on Human Computer Interaction*, IndiaHCI '14, pages 86:86–86:89. ACM, 2014.

[12] C. Mousas. Performance-driven dance motion control of a virtual partner character. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 57–64, March 2018.

[13] T. Takeda, Y. Hirata, and K. Kosuge. Dance step estimation method based on hmm for dance partner robot. *IEEE Transactions on Industrial Electronics*, 54(2):699–706, April 2007.

[14] E. Ho, J. C. P. Chan, T. Komura, and H. Leung. Interactive partner control in close interactions for real-time applications. *ACM Trans. Multimedia Comput. Commun. Appl.*, 9(3):21:1–21:19, July 2013.

[15] I. Rallis, A. Langis, I. Georgoulas, A. Voulodimos, N. Doulamis, and A. Doulamis. An embodied learning game using kinect and labanotation for analysis and visualization of dance kinesiology. In *10th Int. Conf. on Virtual Worlds and Games for Serious Appl. 2018*, pages 1–8.

[16] C. Jennett, A. Cox, P. Cairns, S. Dhoparee, A. Epps, T. Tijs, and A. Walton. Measuring and defining the experience of immersion in games. *Int. journal of human-Computer studies*, 66:641–661, 9 2008.