Creating Virtual Humans with Game Engines for Evaluate Ambient Assisted Living Scenarios

Manuel Sánchez Palacios, Juan Antonio Álvarez-García, Luis Miguel Soria and Damián Fernández Cerero

Abstract In order to evaluate AAL systems, virtual environments help to reduce costs and time, but these environments do not include hyper-realistic human movements. This is something crucial to evaluate activity recognition systems. The present work in progress describes how it is looking for a way to solve this problem and its development using a virtual environment. By means of a game engine and adjusting their parameters, simulations of real acceleration data sets have been generated. It is continuing looking for a valid model to follow.

1 Introduction

Ambient Assisted Living (AAL) is a research area that uses technology to improve the quality of life of elderlies, by increasing their autonomy in daily activities, and by enabling them to feel secure, protected and supported. In order to ensure the effectiveness and usability of AAL approaches, leading to an eventual real-life application, researchers must agree on a set of standard evaluation methods. However, this is a difficult task because of the diversity of both the functions performed by AAL solutions, as well as the heterogeneity of sensors and other hardware used in this field. A function that has emerged as central to many AAL solutions is Activity Recognition (AR), mainly because the ability to understand the user is situation and context are key for real-life usability. Evaluation of AR Systems (ARS) initially was carried out on specific datasets to each system, recorded in laboratory settings, and they generally achieved high accuracy. However, since each research group evaluated their system using their own dataset, a comparison between different ARS was almost impossible. The problem is even worse when the subjects under study are elderly. Most of the current works focused on this section of

M.S. Palacios · J.A. Álvarez-García(✉) · L.M. Soria · D.F. Cerero
Computer Languages and Systems Department, University of Seville, 41012 Seville, Spain
e-mail: {msanpal,jaalvarez,lsoria,damiancerero}@us.es

H. Lindgren et al. (eds.), *Ambient Intelligence- Software and Applications (ISAmI 2016)*,
Advances in Intelligent Systems and Computing 476,
DOI: 10.1007/978-3-319-40114-0_12

the population, make use of synthetic dataset obtaining from younger users imitating their behaviour. The first improvement towards standardized test was achieved when the first benchmark datasets were developed, enabling researchers to compare different ARS. However, for ARS, this approach is not enough because the comparison is limited to systems that use the same configuration, type and placement of sensors, as the one used while recording the dataset.

To tackle this challenging issue, live competitions appeared. The first living competition in ARS was EvAAL-AR [1, 2] (Evaluating AAL Systems through Competitive Benchmarking − Activity Recognition). In these contests, each team is required to bring their own ARS, which is evaluated using criteria in the following features: recognition accuracy, user acceptance, recognition delay, installation complexity, and interoperability with AAL systems. Although the EvAAL-AR competition has been an interesting experience, several issues appeared, including reduced number of participants and longer test desired.

These points focus our attention on virtual environments, where a most extensive evaluation can be carried out. In this paper, we describe the existing virtual environments focused on AAL with humans.

The rest of the paper is organized as follows: Section 2 gives an overview of the state of the art. The work in progress of a game engine with hyper realistic human movements is presented in section 3. Section 4 draws the actual conclusions and next steps.

2 Related Work

The testing and the validation of AAL software systems in real environments require huge efforts in terms of work, time and money. Virtual AAL environments allow simulation of humans with wearable devices, virtual sensors in the environment and, ideally, a physic exact to the real world. They will be an effective solution to speed up the development and the testing phase of AAL systems, by limiting the problems that could arise during the tests in real scenarios. They also tackle the problems of large periods of simulation and multiple people in the environment reducing costs and time.

Some tools are designed for simulating real world, but normally are specific for robotics. In this latter, the movements are generated from automatic or human controlled actions. Humans can be included in these simulators, but their movements are not realistic at all[1]. In the next section, AAL simulators where humans can be included are described.

2.1 AAL Simulators

AAL simulators are not a new concept, Ubiwise[4] in 2002 was one of first approaches. It focuses on the use and analysis of environment models for ubiquitous

[1] https://vimeo.com/90004660

computing systems. People are considered individually on a simulation engine based on Quake II, where real people, acting as players in a game, generate information about their own context, which is captured through simulated sensors.

Another interesting tool is UbiREAL[5]. It lets users intuitively grasp how devices are controlled, depending on temporal variation of contexts in a virtual space. The main contribution of UbiREAL is that it simulates physical quantities (e.g. temperature or humidity) and includes a network simulator. Their human movements are not realistic[2].

Roalter, et al. [6] presented a tool chain to simplify research on Intelligent Environments. The tool chain consists of: a middleware (Robot Operating System tool or ROS), environment edition (SweetHome 3D tool), and simulation and visualization (Gazebo tool). The ROS middleware plays an important role in making interactions possible.

In [7] an AAL virtual simulator is developed build on the top of Modular Robots Open Simulation Engine (MORSE[3]). The authors recreate a virtual home automation environment where a handicapped user moves using a wheelchair. The environment includes a virtual thermometer, RFID sensors on the doors, gas detectors, flowmeters and heart rate and apnea detectors. The application on top of this home automation environment is a virtual caregiver that can detect emergency scenarios. This approach is close to our work but we focus on human movement realism and in the scenario proposed by the authors we cannot check the realism of the human movement (only the one of the wheelchair)[4].

Campillo-Sanchez, et al. work [8,9] advances with Ubik Mobile and PHAT architectures to provide developers with tools to test context-aware services based on smartphones in a simulated AAL environment. Their main contribution is that a real smartphone, and not only an emulator, can be connected to the simulated world to conduct manual and automatic tests. Furthermore PHAT combines a game engine, jMonkeyEngine, with a Multi-Agent Based simulator(MaSON) obtaining realistic human movements and behavior.

From biomechanical research [10] there are specific tools such as AnyBody™[5] that model virtual humans very detailed[6] and include motion capture techniques but normally it does not focus on AAL environments so it is out of this scope.

2.2 Virtual Sensors

Virtual sensors, as opposed to physical sensors, can provide indirect measurements by using other available sensor data, models and knowledge. Raveendranathan et al. [11] proposed a virtual sensor framework in the gait analysis domain for the purpose of enabling real-time activity and posture

[2] https://www.youtube.com/watch?v=57juhBRbIsY

[3] http://www.openrobots.org/morse/doc/stable/morse.html

[4] https://www.youtube.com/watch?v=IBuYZFoKmy8

[5] http://www.anybodytech.com/index.php?id=publications

[6] https://www.youtube.com/watch?v=SmJRkjgpoSg

recognition. vanBeek et al. [12] simulated the time trial of athletes in the Tour de France with various physiological and mechanical models.

Our approach initially is simpler, using physics engine from Unity to create a virtual accelerometer although to more complex sensors another option is to include an Android emulator that includes an accelerometer, gyroscope and magnetometer [9].

3 Tools and Development

Our main goal is to develop an AAL simulator that allows hyperrealist human movements based on real-world physics using a game engine. Once this objective is achieved, measuring acceleration in any part of the model is body (wrist, chest, hip, etc.) will be possible using virtual-wearable sensors. This work in progress describes the steps to achieve this aim. In brief, next stages are involved to develop the virtual environment ecosystem:

1. Select a game engine.
2. Modify the physics engine to obtain real-world physics.
3. Generate a virtual human with enough animation to get hyperrealism.
4. Compare the measurements of our hybrid physics+animation engine with real datasets.

Steps 2 to 4 have been repeated to achieve good results.

The first step was critical to get our objective. Although open source game engines such as JMonkeyEngine, allow the modification of all the code, most popular game engines (Unity, Unreal Engine 4, Source 2 and CryENGINE) are more accepted, their developer community is much greater and introducing latest technical advances quicker. Finally, Unity, a well-known gameengine oriented to 3D games, was selected to model the physics in the AAL environment. These specifications make it a powerful tool to develop the first testing scenario. One of the main advantages of this engine is the possibility to import and export existing formats in a large number of platforms. This will help to share the test environment with other platforms. Furthermore, Unity is the leader in developing gaming platforms, with more than four millions registered developers and it is used in more than 45% of entire market projects for game engines [7], about three times the rate of the nearest competitor.

Developers might encounter issues with physics in Unity, mainly because this game engine is not oriented to real world simulations. By hence, it does not use a metric system. To attach the engine with real world physics units, some adjustments have been performed. It is assumed that 1 unit in Unity is 1 meter in the real environment and the gravity is modified to approach to 9.81 m/s^2. Friction is needed to decrement movement, but Unity is not prepared for ground reaction force, so a modification on the behaviour of the physics materials has been included.

The third point refers to the process of achieving the hyperrealism in humans.In this vein, a mixed model of physics+animation was chosen. The combination of physics and animations is necessary because Unity's physic engine does not include fine-grained movements. By other side, using animation is not suitable to simulate real physics and get distinct reaction in different instances with the same path. The more promising model for animations is Mixamo Test Model [8]. The Mixamo default animations are not as realistic as they should, but with some modifications and simplifications as we can see in Figure 1 and adding a Rag Doll blended with animation the result should be better.

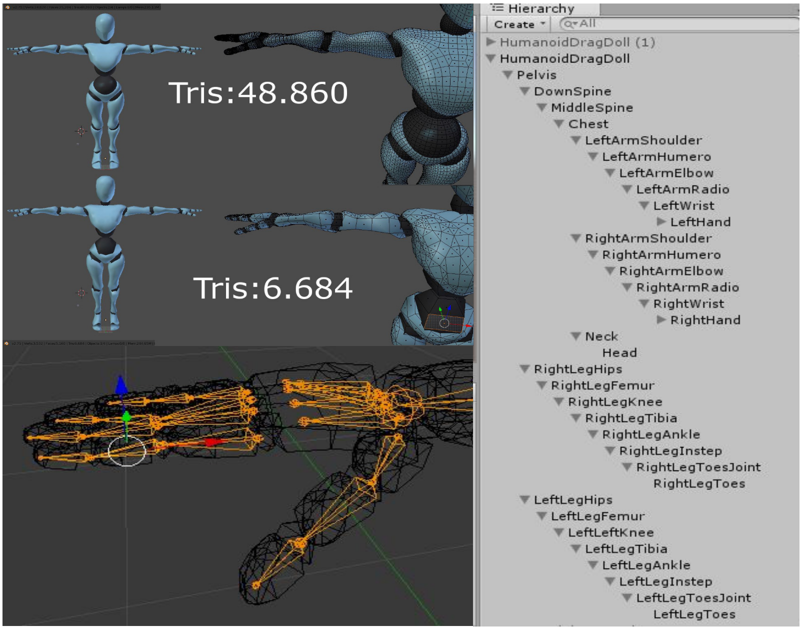

Fig. 1 Simplifying Mixamo model (top left), Including new bones (bottom left), Hierarchy of human Rag Doll (right)

The last step checks how close are the virtualized human actions to real-world physics movements. In order to evaluate the progress, a virtual accelerometer to our character was placed on the scene, performing movements included in a real dataset. The dataset used in this test was UCI Donald Bren [9], which collects accelerometer data from an Android Phone carried on the chest as a sensor. To compare distinct experiments with the dataset, there was established some Key Performance Indicators (KPIs), indicating the accuracy of the reading from the virtual environment compared to the real dataset. Iterating with these KPIs, the current virtual environment approaches to the real data. The KPIs values used were: Maximum acceleration, Minimum acceleration, Curve behaviour, Average acceleration and Histogram of Frequency.

These KPIS are the first minimum KPIs if the model does not get good results with these key numbers we can discard or iterate the model. In the future new more accurate KPIs will be defined to check the data but in the actual state of the project we are checking the environments created viability and the checking is continued during today.

Iterating only an animations environment, the results of the KPIs were better. Once the results were collected, the KPIs were too fixed for a real world environment but some parameters were well-adjusted. The KPIs were acquired again in an absolute-physical model. However, iterating in this profile, the results were unsatisfactory. Finally, now the system is testing with a mixed environment profile, where animations were more accurate aided by real physics. When the final configuration was reached and good KPIs were achieved, a dataset builder with Android smartphones as sensor was developed to ensure that the paths followed by the virtual agents, and the virtualized test are as accurate as the real experiment is.

As we see in the Figure 2 with the same frequency we have much more iteration and noises that in the real sample does not appear. These graphics are extracted by the only animation environment. Here an emulated accelerometer chest sensor where comparing it is KPIs with the UCI data sets. In this environment adjusting the animations good KPIs could be finally reached but every time the same test is reproduced, exactly the same data will be obtained and it is not a real world approach, but is a good beginning for closed and repeated test, which this investigation were not looking for.

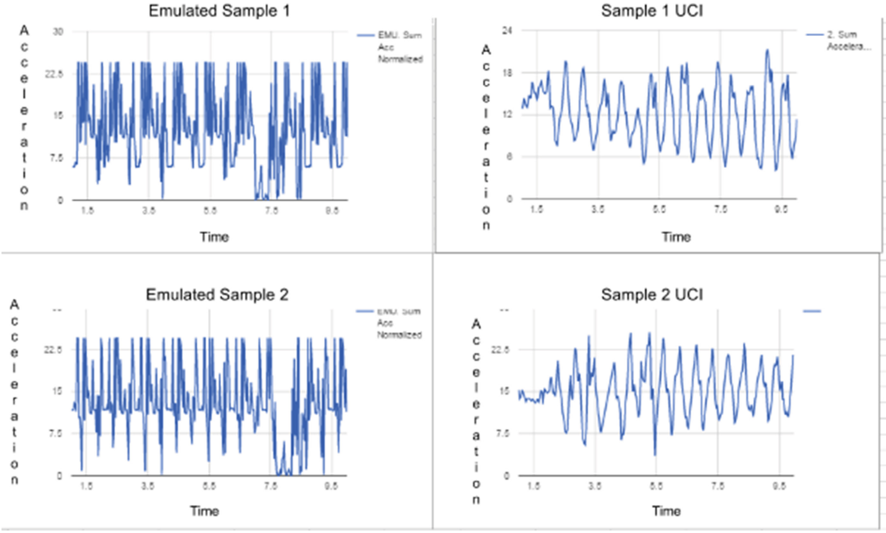

Fig. 2 Comparing UCI Dataset with simulation results.

There are several software tools to model, create, render and execute experiments in real time. We considered SketchUp to model the Living Lab, it is the most arquitectonic one, although there are other options such as SweetHome 3D or Autodesk Homestyler. It allows to easily create models of domestic environments, in which the various sensors and actuators can be placed.

For modelling, rigging and animating the agents, it was used Blender which is a very powerful free software easy to use and with professional performance.

4 Conclusions

In this work in progress, we explained the current trend to use virtual environments to test real solutions in AAL, analysed the state of the art of AAL simulators including humans with realistic movements and showed the steps given to achieve our goal: measure acceleration with wearable virtual devices in any part of our virtual humans.

As future work, we want to include realistic human behaviour such as [13] and create a framework that allows interaction with external researchers such as [14]. This kind of framework could ease focus on specific goals such as [15].

Acknowledgments This research is partially supported by the projects of the Spanish Ministry of Economy and Competitiveness, HERMES (TIN2013-46801-C4-1-r) and Simon (TIC-8052) of the Andalusian Regional Ministry of Economy, Innovation and Science.

References

1. Álvarez-García, J.A., et al.: Evaluation of localization and activity recognition systems for ambient assisted living: The experience of the 2012 EvAAL competition. Journal of Ambient Intelligence and Smart Environments **5**(1), 119–132 (2013)
2. Gjoreski, H., et al.: Competitive Live Evaluations of Activity-Recognition Systems. Pervasive Computing, IEEE **14**(1), 70–77 (2015)
3. Álvarez-García, J.A., et al.: Evaluating wearable activity recognition and fall detection systems. In: 6th European Conference of the International Federation for Medical and Biological Engineering. Springer International Publishing (2015)
4. Barton, J.J., Vijayaraghavan, V.: Ubiwise, a ubiquitous wireless infrastructure simulation environment. HP Labs (2002)
5. Nishikawa, H., et al.: UbiREAL: realistic smartspace simulator for systematic testing. In: UbiComp 2006: Ubiquitous Computing, pp. 459–476. Springer, Heidelberg (2006)
6. Roalter, L., et al.: Developing intelligent environments: a development tool chain for creation, testing and simulation of smart and intelligent environments. In: 2011 7th International Conference on Intelligent Environments (IE). IEEE (2011)
7. Calvaresi, P., et al.: A virtual caregiver for the AAL: Testing in a 3D simulator. In: Proceedings of the 1st Italian Workshop on Artificial Intelligence for Ambient Assisted Living, co-located with the 13th AI* IA Symposium on Artificial Intelligence (2014)

8. Campillo-Sanchez, P., Gómez-Sanz, J.J., Botía, J.A.: PHAT: physical human activity tester. In: Hybrid Artificial Intelligent Systems, pp. 41–50. Springer, Heidelberg (2013)
9. Campillo-Sanchez, P., Serrano, E., Botía, J.A.: Testing context-aware services based on smartphones by agent based social simulation. Journal of Ambient Intelligence and Smart Environments **5**(3), 311–330 (2013)
10. van den Bogert, A.J., et al.: A real-time system for biomechanical analysis of human movement and muscle function. Medical & Biological Engineering & Computing **51**(10), 1069–1077 (2013)
11. Raveendranathan, N., et al.: From modeling to implementation of virtual sensors in body sensor networks. Sensors Journal, IEEE **12**(3), 583–593 (2012)
12. Van Beek, J.H.G.M., et al.: Simulating the physiology of athletes during endurance sports events: modelling human energy conversion and metabolism. Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences **369**(1954), 4295–4315 (2011)
13. Li, Y., Guo, Y.: Wiki-Health: from quantified self to self-understanding. Future Generation Computer Systems **56**, 333–359 (2016)
14. Campuzano, F., et al.: Generation of human computational models with machine learning. Information Sciences **293**, 97–114 (2015)
15. Serrano, E., Poveda, G., Garijo, M.: Towards a holistic framework for the evaluation of emergency plans in indoor environments. Sensors **14**(3), 4513–4535 (2014)