AVATAR: An Open Source Architecture for Embodied Conversational Agents in Smart Environments

Marcos Santos-Pérez, Eva González-Parada, and José Manuel Cano-García

Electronic Technology Department, School of Telecommunications Engineering,
University of Malaga, Teatinos Campus, 29071 Malaga, Spain
{marcos_sape,gonzalez,jcgarcia}@uma.es

Abstract. Due to a growing older population, researchers and industry are paying more attention to the needs of this group of people. Ambient Intelligence (AmI) aims to help people in their daily lives, achieving a more natural interaction of users with an electronic home environment. Embodied Conversational Agents (ECAs) arise as a natural interface between humans and AmI. Our contribution is to present AVATAR: an architecture to develop ECAs based on open source tools and libraries. In the current prototype the virtual agent acts as a natural control interface of the home automation system. In addition, we provide the details to allow its use by Spanish speakers.

1 Introduction

World population ageing is caused by lower overall mortality and fertility. The older population is growing at an enormous pace. In absolute terms the number of older persons has tripled since 1950 and is expected to triple again by 2050 [8].

Older people want to maintain as much independence as possible in their lives. Ambient Assisted Living (AAL) aims to make life easier for people in their own homes by using a set of advanced electronic sensors and automated devices.

A major trend in the development of interfaces for intelligent environments is the use of Embodied Conversational Agents (ECAs) who act as mediators [5]. The daily use of such interfaces creates an illusion of collaboration and may develop social and emotional ties to these virtual assistants. Another interesting application for ECAs is the management of the user's medical needs in their own homes[15]. Research has found that face to face conversation with an embodied agent increases the participation with the system [17]. Furthermore, the presentation of content through the use of a virtual agent improves the ability to retain information [2].

Different architectures for ECAs have been proposed through the integration of various programs and corporate tools [19] [1]. This article focuses on the description of an open source and free architecture for ECAs named AVATAR. The vast majority of research in the field of ECAs assume the use of English between the user and the system but our prototype works in Spanish due to its worldwide distribution.

This paper is organized as follows: after this introduction, Section 2 describes the architecture of our AVATAR platform. The most important aspects of each component of the platform are explained in each subsection. Section 3 summarizes the conclusions drawn from the paper and discusses the future improvements of AVATAR.

J. Bravo, R. Hervás, and V. Villarreal (Eds.): IWAAL 2011, LNCS 6693, pp. 109–115, 2011.

2 AVATAR Platform Architecture

In this section we describe the actual implementation of the AVATAR platform. Figure 1 shows its block diagram. It consists of the following basic components:

- *Voice Activity Detector (VAD)* discriminates voice against environmental noise.
- *Automatic Speech Recognition (ASR)* performs speech to text conversion.
- *Conversational Engine (CE)* extracts the meaning, controls the dialog flow and produces a semantic representation suitable for the task context. It generates a response based on the input, the current state of conversation and dialog history.
- *Task Manager (TM)* connects to the environmental devices.
- *Text-To-Speech (TTS)* generates the output speech signal and the timing configura tion for the virtual head animation.
- *Virtual Head Animation (VHA)* sets the facial animation synchronized with the output speech.

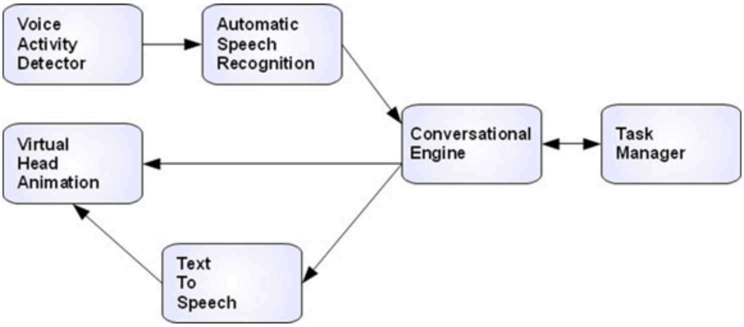

Fig. 1. AVATAR architecture

The architecture follows a modular design so that each component of the platform can be modified without affecting others. Each component runs in a different thread and communicates with the next by a message queue.

2.1 Voice Activity Detector

The VAD role is to differentiate the audio segments where the user speaks from those containing noise. VAD are usually used in other scenarios, such as mobile communications and Voice over IP (VoIP). In these scenarios, the VAD function aims to achieve a reduction of network traffic in order to save bandwith. Although the operating principles are the same, the purpose of the VAD in conversational interfaces is to segment the user's speech into sentences.

The VAD is fed directly with the digitized samples coming from the microphone and sends the raw audio from the segments containing user's voice to the ASR.

The most common VAD algorithms are based on energy threshold and zero crossing rate of the signal. In our platform this role is performed by the SphinxBase library [7].

2.2 Automatic Speech Recognition

Voice recognition is a crucial part of the conversational system and its functionality is essential to allow voice communication between human beings and electronic systems.

The complexity of such systems lies in the diversity of factors that includes human speech (acoustics, phonetics, phonology, lexicon, semantics). In many cases, the sense of naturalness of the conversational interface depends heavily on the robustness of speech recognition [14]. Despite these difficulties some notable advances in this field have been achieved in recent years and make possible automatic speech recognition with acceptable error levels for a large number of applications [23].

The recognizer can be viewed as a black box that transforms the voice segments that come from the VAD directly to text, which is sent to the CE.

In our AVATAR platform, speech recognition is performed by PocketSphinx library, which belongs to the CMU Sphinx family [6]. The reasons for this choice are that this library allows speaker-independent speech recognition and has been used in real-time applications [12].

Voice recognition needs both an acoustic and a language corpus to run. For this project, we used the Voxforge Spanish corpus [21]. It is the first free speaker-independent acoustic model for spanish. The statistical language model in the prototype is generated from a set of possible sentences that led to a dictionary of 75 different words to control the various elements of the simulated environment.

2.3 Conversational Engine

Classic conversational agents usually lead to too linear and rigid talks. For a more natural feel and in order to avoid being rejected by users, it is necessary that the system can handle unexpected changes of context and be able to achieve many goals asynchronously [13]. One approach to this type of systems with great success in both academic and commercial sector is that of conversational bots.

A chatbot or conversational bot is a program that simulates a conversation with someone. There are currently a large number of chatterbots or chatbots used in various fields [16] such as marketing, education [4] or entertainment. One of the most remarkable is ”ELIZA” [22], which is considered the forerunner of the current conversational bots. The most successful chatbots seem to be based on AIML, an XML-based language that is considered a de facto standard after its composition and operation were published.

The conversational engine of AVATAR platform is based on PyAIML [20]. It is an open source AIML interpreter and it is fully compatible with AIML 1.0.1 standard.

The CE module receives the text from the ASR and draws its possible meaning. In our prototype its main objective is to obtain the values that fill the slots needed for the TM. These slots correspond to the action to take, the type of object that receives the action, the name of the object, the change of the object state and the room where the object is. In addition, the CE module must also manage the context and history of the conversation since the user is free to provide all necessary data at one time or not. With this information the CE module communicates with the TM and generates a textual response understandable to the user informing him the result of his action or requesting for further information.

2.4 Task Manager

The task manager provides an interface between the conversational engine and the smart home.

While the ECA can run on a handheld device, the control system of the house is usually implemented in a desktop computer. Therefore, the task manager provides a client-server architecture. The client module runs on the same machine as the ECA and then sends requests to the home. On the other hand, the server module translates the requests into the language of the home automation system and responds with confirmation or error messages or to the agent.

In the current development stage of the AVATAR project automation control system is simulated. Figure 2 graphically displays the status of various objects that can be managed on the server.

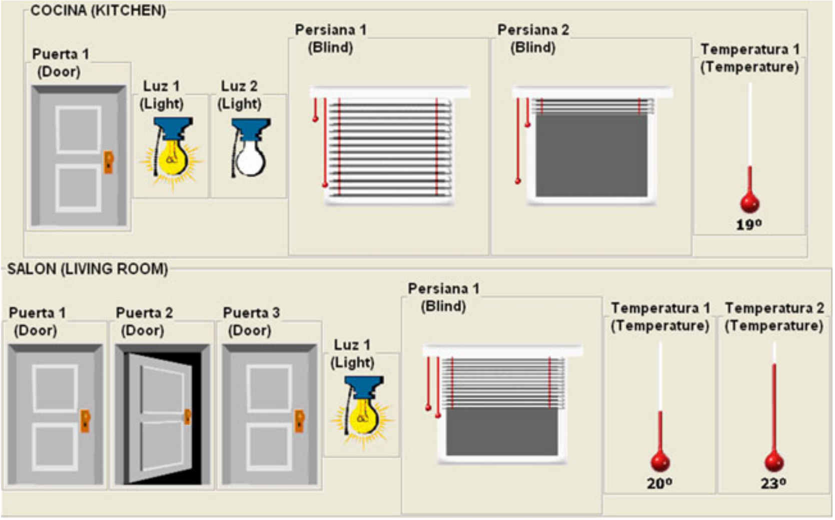

Fig. 2. Simulated devices controlled on the server

2.5 Text-To-Speech

There are different methods for obtaining a synthetic voice and each has advantages and disadvantages. Concatenative synthesis is achieved by attaching pre-recorded voice segments. For specific application domain it is often preferred to store whole words, which leads to a high quality artificial voice. Instead formant synthesis produces the sound wave varying physical parameters of an acoustic model. By not using human voice segments, it produces a robotic voice and its usage only makes sense in systems with limited hardware resources. There are other types of synthesis, but they are not yet matured enough for its use in complete systems.

The TTS module just needs the response text from the CE to synthesize the artificial voice. Just before starting the playback of the response it sends the data with the duration of each vocal segment to the VHA module.

The TTS system used in AVATAR is Festival [10]. Festival TTS is an open source synthesizer of free use from the University of Edinburgh. Festival offers a robust synthesis algorithm based on concatenation of diphonemes. It currently supports both English and Spanish.

The default voice in Spanish of Festival does not offer a high quality, so we opted to use the male voice from the Hispavoces project of the Andalusian Regional Government [11].

2.6 Virtual Head Animation

The virtual head is the embodiment of intelligent agent. A usual rule of avatar design states that the realism of its appearance must correspond with its behavior. Otherwise, the avatar will be rejected because of what is known as the Uncanny Valley effect [9]. For this reason an effort was made to give it the greatest possible realism. The head model includes 22 different bones and has more than 30 different animated expressions in addition to the 10 visemes that are needed to simulate speech.

The response text from CE and the duration of the audio segments from the TTS module are the needed entries for the animation of the virtual head. The technique used to achieve lip synchronization is as follows. The first step is to conduct a preliminary analysis of the response text generated by the conversational engine in order to obtain the list of visemes to be executed simultaneously with the audio. The second step is to modulate the playback speed of each visema animation depending on the length of each segment of the sentence. It is worth noting that the selection of facial expressions during the speech animation is still under development.

Head modeling and animation were made with Blender [3]. Blender is a multiplatform software application that specializes in 3D graphics modeling and animation. Released as free software under the GNU General Public License, Blender is available for a number of operating systems, including GNU/Linux, Mac OS X, FreeBSD, OpenBSD and Microsoft Windows.

OGRE [18] is used as rendering engine in AVATAR. It is a scene-oriented 3D rendering engine written in C++. The class library abstracts the details of using the underlying system libraries like Direct3D and OpenGL and provides an interface based on world objects and other high-level classes. Released under the terms of the MIT License, the engine is free software.

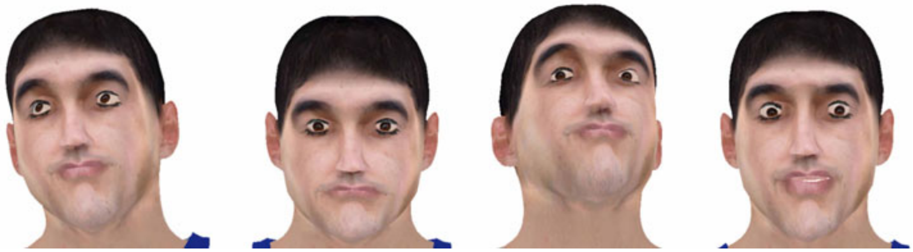

Fig. 3. Virtual head in different poses

3 Conclusion and Future Work

The main goal of this work was to describe a software platform aimed at developing ECAs. Thus, we proposed a possible design and defined the architecture and implementation details for such platform. We made an extra effort during the election of the components to be integrated so we could obtain a free and open source platform.

Currently, the platform is still under development, so hopefully the final version will present improvements over the version described in this document.

A line of future work focuses on modeling users through Case-Based Reasoning (CBR). In order to automatically identify the speaker it can be used a classic technique of pattern matching based on the features of the user's voice.

A second line of future work is related to the integration of the whole platform in an embedded system in order to use it in new user devices like mobile phones or tablet computers.

Acknowledgments

This work was partially supported with FEDER/MCINN funds by the National Project TEC2009-13763-C02-01 and by the Andalusian Regional Project P08-TIC4198.

References

1. Baldassarri, S., Cerezo, E., Serón, F.: An open source engine for embodied animated agents. In: XVII Congreso Español de Informática Gráfica (CEIG 2007), Zaragoza, Spain, pp. 91–98 (September 2007)
2. Beun, R.-J., de Vos, E., Witteman, C.: Embodied conversational agents: Effects on memory performance and anthropomorphisation. In: Rist, T., Aylett, R.S., Ballin, D., Rickel, J. (eds.) IVA 2003. LNCS (LNAI), vol. 2792, pp. 315–319. Springer, Heidelberg (2003)
3. Blender website, `http://www.blender.org`
4. Burguillo-Rial, J.C., Rodríguez-Silva, D.A., Santos-Pérez, M.: T-Bot: an intelligent tutoring agent for open e-Learning platforms. In: 8th International Conference on Information Technology Based Higher Education and Training (ITHET 2007), Kumamoto City, Japan (July 2007)
5. Carolis, B.D., Mazzotta, I., Novielli, N., Pizzutilo, S.: Social robots and ECAs for accessing smart environments services. In: Proceedings of the International Conference on Advanced Visual Interfaces, AVI 2010, pp. 275–278. ACM, New York (2010)
6. CMU Sphinx website, `http://cmusphinx.sourceforge.net/`
7. CMU Sphinxbase website, `http://sourceforge.net/projects/cmusphinx/`
8. D. of Economic and Social Affairs. Population Division. World population ageing 2009. Tech. rep., United Nations
9. Fabri, M.: Emotionally Expressive Avatars for Collaborative Virtual Environments. PhD thesis, Leeds Metropolitan University, Leeds, UK (November 2006)
10. Festival Speech Synthesis System website, `http://www.cstr.ed.ac.uk/projects/festival/`
11. Guadalinex Hispavoces website, `http://forja.guadalinex.org/frs/?group_id=21`

12. Huggins-daines, D., Kumar, M., Chan, A., Black, A.W., Ravishankar, M., Rudnicky, A.I.: PocketSphinx: a free, real-time continuous speech recognition system for hand-held devices. In: Proc. of ICASSP, Touluse, France, pp. 185–188 (May 2006)
13. Hung, V., Gonzalez, A., Demara, R.: Towards a Context-Based dialog management layer for expert systems. In: International Conference on Information, Process, and Knowledge Management, eKNOW 2009, Cancun, Mexico, pp. 60–65 (February 2009)
14. Jokinen, K.: Natural language and dialogue interfaces. In: The Universal Access Handbook, 1st edn., pp. 495–506. CRC Press Taylor & Francis Group (2009)
15. Kenny, P., Parsons, T., Gratch, J., Rizzo, A.: Virtual humans for assisted health care. In: Proceedings of the 1st international conference on PErvasive Technologies Related to Assistive Environments, PETRA 2008, pp. 6:1–6:4. ACM, New York (2008)
16. Kowalski, S., Pavlovska, K., Goldstein, M.: Two case studies in using chatbots for security training. Brazil (2009)
17. Mulken, S.V., André, E., Müller, J.: The persona effect: How substantial is it? In: Proceedings of HCI on People and Computers XIII, HCI 1998, pp. 53–66. Springer, London (1998)
18. OGRE website, `http://www.ogre3d.org`
19. Pejsa, T., Pandzic, I.S.: Architecture of an animation system for human characters. In: Proceedings of the 10th International Conference on Telecommunications, ConTEL 2009, pp. 171–176. IEEE, Zagreb (2009)
20. PyAIML. A Python AIML Interpreter website, `http://pyaiml.sourceforge.net/`
21. Voxforge Spanish Model website, `http://cmusphinx.sourceforge.net/2010/08/voxforge-spanish-model-released/`
22. Weizenbaum, J.: ELIZA: computer program for the study of natural language communication between man and machine. Commun. ACM 9(1), 36–45 (1966)
23. Zhang, J., Ward, W., Pellom, B., Yu, X., Hacioglu, K.: Improvements in audio processing and language modeling in the CU communicator. In: Eurospeech 2001, Aalborg, Denmark (September 2001)