ECA-based Control Interface on Android for Home Automation System

Marcos Santos-Pérez, Eva González-Parada and José Manuel Cano-García
School of Telecommunications Engineering, University of Malaga, 29071 Malaga (Spain)
Email: marcos_sape@uma.es, eva@dte.uma.es, cano@dte.uma.es

***Abstract*—Historically, Embodied Conversational Agents (ECAs) have been used as virtual assistants that make easier the access to information or help in performing complex tasks. Due to their high computational requirements ECAs are usually run on desktop computers, but with the recent development of hand-held devices both in hardware and software, it becames neccessary to move ECAs to that new mobile scenario. Thus, we propose an open-source based platform for developing ECA based interfaces on Android-equipped devices. We also present a prototype for controlling a home automation system.**

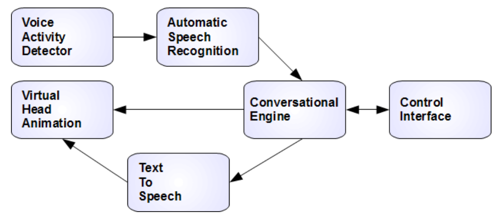

Fig. 1. Architecture of the proposed platform.

I. Introduction

Embodied Conversational Agents (ECAs) are animated virtual characters that emulate human behaviour and communication. ECAs arise as a conversational partner for the user in computer-based environments [1]. Instead of addressing the computer, the user addresses a virtual agent that can be made responsible for certain tasks, as performing a web search, answering a fixed-domain question or controlling the home automation system [2] among others. Other approach is the use of an ECA as a sociable and emotionally intelligent companion for the user [3].

Due to the limited computational power of hand-held devices compared to desktop computers, the most common architectures for ECA-based mobile applications rely on an external server that performs the processor intensive tasks, such as speech recognition, language understanding and text-to-speech [4]. But in the last few years the embedded processors have significantly increased their computational power and it starts to be possible to run an ECA completely within a hand-held device.

This paper describes an platform for developing ECA-based interfaces on Android hand-held devices. The proposed platform is based on free and open source libraries. We developed a prototype installed on a tablet for controlling a home automation system.

II. ECA Platform Overview

The architecture of the platform shown in Figure 1 follows a modular design so that each component can be modified without affecting others. The conversational engine is implemented as a Python module and the rest of the elements are provided as native libraries, accessible through a facade-interface.

A. Voice Activity Detector

The Voice Activity Detector's (VAD) role is to discriminate the user's voice frames from those containing noise. That way, the VAD allows the segmentation of the user's speech into utterances. This module reads the digitized audio samples acquired from a microphone and sends the filtered raw audio to the ASR. The actual implementation of the VAD module is based on the SphinxBase library, which was modified so it can work with the OpenSL ES native audio libraries present on Android.

B. Automatic Speech Recognition

The Automatic Speech Recognition (ASR) module performs speech to text conversion. It takes as input the utterance with the user's speech that come from the VAD and send the resultant text to the CE. In the proposed platform, the ASR module is based on the PocketSphinx speech recognition library. Some changes were made to the original code in order to improve the response time on embedded devices by starting the recognition phase once the first speech frame is detected by the VAD [5] and by choosing the most appropriate language model for the topic of the conversation [6].

C. Conversational Engine

The Conversational Engine (CE) extracts the meaning of the utterance, manages the dialog flow and produces the actions appropriate for the target domain. It generates a response based on the input, the current state of the conversation and the dialog history. The CE module is based on PyAIML, an AIML chatbot. AIML (Artificial Intelligence Markup Language) is an extension to XML that provides symbolic reduction, recursion, context-awareness and history management in order to understand the user's utterance and generate an appropriate response. The original code was improved with an optional lemmatizer submodule that reduces both the response time and the memory usage of the CE module when dealing with inflectional languages. It was also added support for an object-oriented database that can decrease the dynamic memory usage at the expense of an increment of the response time [7].

978-1-4673-1363-6/13/$31.00 ©2013 IEEE

D. Control Interface

The Control Interface translates the commands said by the user to a format that can be understood by the target applications or services running on the same device or accessible remotely. This module is domain-specific and has to be re-implemented or adapted for every new target application.

E. Text-To-Speech

The Text-To-Speech (TTS) subsystem carries out the generation of the synthetic output voice from the text that comes as a response from the CE. For the sake of getting a realistic ECA, it sends to the VHA module a list of the phonemes with their duration so animation and artificial speech match up. The TTS module implementation is based on the eSpeak library.

F. Virtual Head Animation

The virtual head is the embodiment of the conversational agent and the visual counterpart of the TTS module.

This module receives as inputs both the mood information from the CE and the list of the phonemes' durations from the TTS module. By processing the inputs, it generates the visemes (the visual representation of the phonemes) and the facial expression that will be rendered along with the synthetic voice. The purpose of the phonemes' timing list is to modulate the animation speed to achieve perfect lip synchronization. OGRE 3D was used as the rendering engine of the software platform.

III. Prototype Details

The ECA requires a version of Android higher than 4.0 (also named ICS-Ice Cream Sandwich) and a hardware capable of coping with the computational tasks that form the platform. For our prototype we used a tablet with an OMAP 4460, which consists of a 1.2GHz dual-core ARM Cortex-A9 as CPU and a PowerVR SGX540 384MHz GPU. Fig. 2 shows the ECA running on the tablet.

The home automation system was previously implemented in a room and has a tactile manual control interface that is showed in Fig. 3. Likewise the manual control interface, the ECA can control the door lock, the lighting, the blinds and the temperature.

Fig. 2. ECA prototype in happy mood.

Fig. 3. Manual control interface of the home automation system.

IV. Conclusions and Future Work

The main goal of this work was to describe a platform aimed at developing ECA-based interfaces on hand-held devices equipped with Android. Thus, we proposed a possible architecture and gave implementation details for such platform. The whole platform is based on free and open source libraries and a first prototype was developed for controlling a home automation system.

The future work consists of to convey some experiments with real users to measure the usefulness, usability and performance of the platform.

Acknowledgment

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

References

[1] M. M. Louwerse, A. C. Graesser, D. S. McNamara, and S. Lu, "Embodied conversational agents as conversational partners," *Applied Cognitive Psychology*, vol. 23, no. 9, pp. 1244–1255, 2009.

[2] B. De Carolis, I. Mazzotta, N. Novielli, and S. Pizzutilo, "Social robots and ECAs for accessing smart environments services," in *Proceedings of the International Conference on Advanced Visual Interfaces*, ser. AVI '10. New York, NY, USA: ACM, 2010, pp. 275–278.

[3] M. Cavazza, R. Santos de la Cámara, M. Turunen, J. Relaño Gil, J. Hakulinen, N. Crook, and D. Field, "How was your day?' An Affective Companion ECA Prototype," in *Proceedings of the SIGDIAL 2010 Conference*, Association for Computational Linguistics. Tokyo, Japan: Association for Computational Linguistics, September 2010, p. 277–280.

[4] H.-J. Oh, C.-H. Lee, M.-G. Jang, and K. Y. Lee, "An Intelligent TV interface based on Statistical Dialogue Management," *Consumer Electronics, IEEE Transactions on*, vol. 53, no. 4, pp. 1602 –1607, nov. 2007.

[5] M. Santos-Pérez, E. González-Parada, and J. M. Cano-García, "Efficient Use of Voice Activity Detector and Automatic Speech Recognition in Embedded Platforms for Natural Language Interaction," in *Highlights in Practical Applications of Agents and Multiagent Systems*, ser. Advances in Intelligent and Soft Computing. Springer Berlin Heidelberg, 2011, vol. 89, pp. 233–242.

[6] M. Santos-Pérez, E. González-Parada, and J. Cano-García, "Topic-Dependent Language Model Switching for Embedded Automatic Speech Recognition," in *Ambient Intelligence - Software and Applications*, ser. Advances in Intelligent and Soft Computing. Springer Berlin / Heidelberg, 2012, vol. 153, pp. 235–242.

[7] M. Santos-Pérez, E. González-Parada, and J. M. Cano-García, "Embedded Conversational Engine for Natural Language Interaction in Spanish," in *Proceedings of the Paralinguistic Information and its Integration in Spoken Dialogue Systems Workshop*. Springer New York, 2011, pp. 365–374.