Design Considerations of an Interactive Robotic Agent for Public Libraries

Stelios M. Potirakis\textsuperscript{a}, Todor Ganchev\textsuperscript{b}, Gürkan Tuna\textsuperscript{c,*}, Nicolas-Alexander Tatlas\textsuperscript{a} and Recep Zogo\textsuperscript{d}

\textsuperscript{a}Department of Electronics Engineering, Technological Education Institute of Piraeus, Aigaleo, Greece
\textsuperscript{b}Department of Electronics, Technical University Varna, Bulgaria
\textsuperscript{c}Department of Computer Programming, Edirne Vocational School of Technical Sciences, Trakya University, Edirne, Turkey
\textsuperscript{d}Directorate of Library and Documentation, Trakya University, Edirne, Turkey

\* Corresponding author. Tel.: +90-284-224-0283; fax: +90-284-224-0287; e-mail: gurkantuna@trakya.edu.tr

I. Introduction

Public libraries are libraries that are accessible by the general public and are operated by civil servants. They exist in many countries across the world and are often considered an essential part of having an educated and literate population (Rubin, 2010). In public libraries, users are allowed to take books and other materials off temporarily. In order to improve service quality of libraries, users should be guided well by the civil servants and computer-based applications (Hsieh, Chang and Lu, 2000).

Library service research has recently focus on the exploitation of different forms of concurrent technological facilities in order to improve the services offered to typical, distant, as well as to physically or visually impair users (Breitbach and Prieto, 2012), (Cho, Kim, and Cha, 2012), (Evans and Reichenbach, 2012), (Fassbender and Mantora, 2013), (Hill, 2013), (Jonnalagadda, 2012), (Mallon, 2012), (Mikawa, Morimoto and Tanaka, 2010), (Singh and Moirangthem, 2010). These technological facilities include smart-phones, internet access, robots and voice interaction with the user.

Human-Robot Interaction (HRI) is an extensive research area covering several topics towards direct interaction with robots through gesture and speech (Kiesler and Hinds, 2004). It plays a key role in the development of socially-intelligent robots by enabling them to understand the requirements of their users (Yanco and Drury, 2004). It is used in several robotic applications such as teleoperation, robot hosting, and service robots (Werner, Oberzaucher and Werner, 2012). HRI-based applications are realized in three main steps: 1- awareness and acquisition, 2- interpretation and 3- execution.

In this study, a robotic agent for serving users is proposed. By recognizing speech, the agent determines tasks to be executed and is supposed to aid civil servants in guiding users. Supported by automatic speech recognition (ASR) and text to speech (TTS) synthesis systems, the agent can understand several commands and ask clarifying questions in case of ambiguity.

The rest of the paper is organized as follows. Section II presents the details of the robotic agent designed to operate in libraries. Section III gives the details of the speech interaction subsystems of the proposed system. Finally, the paper is concluded in Section IV.
II. System Architecture

The architecture of the proposed system is shown in Fig. 1. The system translates the speech of the users into text and then carries out tasks given to. On the other hand, any feedback given by the system to the user is first generated in text form and then transformed to synthesized speech.

Each component of the proposed system is briefly described below.

- Audio stream: Utterances from users.
- Speech to text: Translates the utterances into text using an ASR subsystem.
- Dialog management: Manages dialog and resolves any ambiguities through clarifying questions.
- Task Management – Determines tasks to be executed.
- Audio feedback: Generates any feedback messages from the robotic agent in text form and then synthesizes speech using a TTS subsystem.

In applications, such as library robot agent the task completion rate criterion is adopted. A task is considered completed when the recognition engine provides a correct transcription of the speaker’s intention. The speech understanding component parses every command phrase and translates it to the corresponding concept, which feeds the dialog generation module. The dialog generation flow operates on a conceptual level, which facilitates the handling of the variety of library user requests.

![Fig. 1. An overview of the system architecture](image)

III. Speech Interaction Subsystems

Recently, speech recognition and synthesis have become key techniques in intelligent HRI. Speech interaction, which is the fundamental way of communication among humans, is not just an easy way to interact with the machine; especially in the case of visually impaired people and in general people with physically disabilities that do not permit them to use mouse or keyboard as input devices this is probably the most effective way to communicate with the machine (Sharma and Wasson, 2012).

Command and Control Speech Recognition

In applications with speech-enabled HRI it is extremely important to guarantee reliable and error-free communication. This is the reason why the speech interface in robot agents is typically implemented as a command and control speech recognition. In such cases, the user interface makes use of a relatively small set of carefully chosen commands, which facilitates the use of strict grammars and boosts the speech recognition accuracy. In fact, in command and control speech recognition in indoor environments, it is typical to achieve nearly perfect speech recognition accuracy for vocabulary size of fifty, sixty or even larger number of command phrases (word combinations). Due to the use of strict grammars even when the word recognition rate is not perfect, the task completion rate is close to 100%. These considerations motivated our choice of command and control speech recognition for the speech interface of the library robot.

The robotic agent ASR subsystem consists of two modules: (a) the word spotter and (b) the command and control speech recognition module. The word spotter is constantly seeking a specific word (e.g. Agent), which indicates that what follows is a command to the robotic agent. Detecting the word Agent activates the command and control speech recognition module. Thus, instructions which are not preceded by the word Agent are ignored. This rule follows the good practices of command and control speech interfaces, a safety measure so that the robot is not activated sporadically during discussion between people.

For the development of the word spotter and the command and control speech recognition module we made use of the Microsoft Speech API 5.3, which provides a high-level interface to a speech recognition engine (MSDN Microsoft, 2013a). Specifically, we set the speech recognizer in shared recognition engine mode, so that the word spotter and the command and control speech recognition use the same instance of the engine but with different grammars. This allows for more efficient use of the hardware resources, as the word spotter and the command and control speech recognition components do not work simultaneously, but are cascaded. The word spotter enables the command and control speech recognition module only if the robot name Agent is included in the command phrase.

For the command and control speech recognition module, we established a simple grammar, which can be summarized as:

\[ \text{Agent} \rightarrow (\text{please}) \rightarrow \{\text{verb}\} \rightarrow (\text{a | an | the}) \rightarrow \{\text{item}\} \]

where the set of allowed actions is \(\{\text{verb}\} = \text{find | bring | go | show | …}\) and the set of allowed objects is \(\{\text{item}\} = \text{article | journal | book | …}\). Any word which does not fit this grammar is ignored. For safety reasons, when the confidence level of the speech recognizer is low the robotic agent asks (through the text-to-speech module) the
command to be repeated.

**The Text to Speech Subsystem**

Speech synthesis is the artificial production of human speech, as natural and intelligible as possible. In the special case that a speech synthesis subsystem accepts text as input and narrates it to speech, this subsystem is referred to as a text-to-speech synthesizer or TTS for short. TTS is the function that is complementary to ASR; a robot having both ASR and TTS subsystems can provide a full speech-based HRI (Holmes and Holmes, 2001).

Fig. 2 provides the block diagram of a generic TTS module. In a higher level this can be divided into two parts (Van Santen et al., 1997), the “front-end” providing a translation of the input text to convert the input text to sound intermediate linguistic representation, and the “back-end” producing the synthesized speech waveform based on the obtained linguistic representation. The front-end first processes the provided character strings, e.g., identifies numbers, abbreviations, resolves different spellings for a word, etc., to identify the spoken words corresponding to the provided text. Then, then assigns phonetic transcriptions to each word, and finally, the linguistic analysis follows which reveals linguistic parameters, like the phonetic values of the parts of each word, divides the text into prosodic units, like phrases, clauses, and sentences and finally defines prosody characteristics, i.e., duration, intonation (involving fundamental frequency $F_0$, i.e. pitch, information) and intensity patterns of speech for the sequence of syllables, words and phrases (Van Santen et al., 1997), (Reddy and Rao, 2013).

The quality of TTS subsystem is determined by the similarity of the synthesized voice to human voice (naturalness), and the extent to which the synthesized voice is understood (intelligibility). Provided that a successful front-end has been designed for the language of interest, the quality of the TTS subsystem is determined by the success of the back-end, i.e., the method of waveform synthesis. Therefore, it has been a subject of intense research during last decades.

The early TTS back-ends (until the 1990’s) were using articulatory or sinewave formant synthesis (e.g. (Rubin, 1982), (Remez at al., 1981)). However, due to the low naturalness of the synthesized voice, scientists turned to concatenative speech synthesis which is based on the concatenation of segments of recorded speech. These segments may vary from sentences and phrases to diphones and phones (Olive, 1997). This kind of synthesis generally provides better naturalness and intelligibility, however calls for a large recorded speech database (corpus-based) and greatly depends on the segment type (phonetic unit) employed. Although, concatenative synthesis is still active and continuously improving, a more efficient (at least in the required database point of view) formant synthesis method emerged in mid-1990’s, the corpus-based statistical parametric synthesis. This method became quite popular during the 2000’s and is usually found as the HMM-based speech synthesis (Dutoit, 1997), (Ling, 2012). The main advantage of this approach is its flexibility in changing speaker identities, emotions, and speaking styles (Tokuda et al., 2013). It is comprised of two parts: (i) the corpus-based HMM training part, which is very similar to the corresponding HMM ASR training, and (ii) the synthesis part that takes into account the training part and the front-end linguistic parameters to synthesize speech corresponding to the input text.

A number of different TTS products are already commercially available for different languages, while others other TTS engines are free-accessible, some of them being multilingual providing a platform for the development of TTS for different languages. Of course, all operating systems, as well as internet applications provide speech synthesis features.

In this context, Microsoft Speech Application Programming Interface (SAPI) is an API developed by Microsoft to allow the use of speech recognition and speech synthesis within Windows applications. It is possible for a 3rd-party company to produce their own Speech Recognition and Text-To-Speech engines or adapt existing engines to work with SAPI (Chungurski, Arsenovski and Gjorgjevikj, 2012), (Microsoft Research, 2013), (MSDN Microsoft, 2013a). The System.Speech.Synthesis namespace can be used to access the SAPI synthesizer engine to render text into speech using an installed voice, such as Microsoft Anna (Sharma and Wasson, 2012), (MSDN Microsoft, 2013b).

For the purposes of this research we have used a Matlab-based TTS engine that employs SAPI. An example of the obtained speech from an indicative response message that the proposed system should produce after the selection of the requested book by a person that uses the library services is presented in Figs 3 and 4. For the blue parts of the synthesized waveform (Fig. 3, middle panel) that have been identified as possibly voiced (Fig. 3, top panel), the corresponding blue parts of the bottom panel of Fig. 3 show the estimated pitch (fundamental frequency, $F_0$) (Gonzalez and Brookes, 2011), while Fig. 4 shows the spectrogram of the synthesized voice.
IV. Conclusion

Generally, public libraries are in service for several hours. Even if it seldom happens, foreign users may want to use the public libraries. In this situation, an English-speaking public servant is needed to help them. In this study, a robotic agent is proposed to deal with this situation.

The robotic agent consists of an automatic speech recognition system and a text to speech subsystem. A Matlab-based text to speech engine employing Microsoft Speech Application Programming Interface has been used to test the applicability of the proposed robotic agent.

Future work of this study consists of the implementation of Turkish language packages for field tests that will be held at the library and documentation center of Trakya University, Edirne, Turkey.

Acknowledgements

An earlier version of this paper has been accepted for publication in the proceedings of the 7th International Advanced Technologies Symposium (IATS’13).

The authors wish to thank the anonymous reviewers for their valuable suggestions.

References


Journal of Balkan Libraries Union


Authors’ Biographies

Stelios M. Potirakis currently serves as an Assistant Professor at the Department of Electronics of the Technological Education Institute of Piraeus. He received his B.S. degree in Physics with First Class Honors (1993), an M.Sc. degree in Electronics and Communications (1995), and a Ph.D. in Physics (2002), all from University of Athens. Dr. Potirakis has authored several papers in international conference proceedings and refereed journals and has been serving as a reviewer for international journals and conferences. His current research interests include analog electronics, analog circuit analysis, systems’ modeling, complex systems’ signal analysis, electroacoustics, applied acoustics, acoustic ecology, distributed sensor networks for environmental monitoring, electronic measurements, characterization of smart multi-function materials, and educational technologies.
E-mail: spoti@teipir.gr

Todor Ganchev received the Diploma Engineer degree in electrical engineering from the Technical University Varna, Bulgaria, in 1993 and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Patras, Greece, in 2005. From February 1994 to August 2000, he consequently held engineering, research, and teaching staff positions at the Technical University Varna. Between September 2000 and September 2012, he has been a Researcher at the Wire Communications Laboratory, University of Patras, Greece. Since October 2012 he is a Faculty member of the Department of Electronics at the Technical University Varna, Bulgaria. Todor Ganchev authored/co-authored over 120 scientific publications in the areas of bioacoustics, signal processing, speech and audio processing, pattern recognition, and signal processing applications. Todor Ganchev is a Member of the EURASIP and a Senior Member of the IEEE.
E-mail: tganchev@tu-varna.bg

Gürcan Tuna serves as an Asst. Prof. at Trakya University, Edirne, Turkey. He received his B.S. degree in computer engineering from Yıldız Technical University, Istanbul, Turkey, in 1998, and his M.S. degree in computer engineering from Trakya University, Edirne, Turkey, in 2009. He received his Ph.D. degree in electrical engineering from Yıldız Technical University, Istanbul, Turkey, in 2012. Tuna has authored several papers in international conference proceedings and refereed journals, and two book chapters. He has been serving as a reviewer for international journals and conferences. His current research interests include smart grid, ad hoc and sensor networks, robotic sensor networks, multi sensor fusion, energy harvesting, and energy-aware routing.

Nicolas-Alexander Tatlas received an engineering degree in 2001 and a Ph.D. degree in 2006, both from the Department of Electrical and Computer Engineering, University of Patras, Greece. His research focused mainly on digital audio delivery over PANS and WLANs, digital audio amplification, and direct digital electroacoustic transduction. He was also involved in a number of international and national projects in digital audio technology. He has authored and presented more than 20 papers in scientific journals and international conferences. Dr. Tatlas is a member of the Technical Chamber of Greece, an associate member of the Audio Engineering Society, and a committee member of the AES Greek Section. Dr. Tatlas currently serves as a Lecturer at the Dept. of Electronics Engineering, Technological Education Institute of Piraeus, Egaleo, Greece. E-mail: ntatlas@teipir.gr

Recep Zogo received his master's degree in public administration from Trakya University, Edirne, Turkey, in 2009. He currently serves as the Head of Library and Documentation Directorate, Trakya University, Edirne, Turkey. E-mail: recepzogo@trakya.edu.tr