Intelligent Speech Interactive Agent on a Car Navigation Environment Using Embedded ASR and TTS

Heungkyu Lee[1)] and Hanseok Ko[2)]
[1)]Dept. of Visual Information Processing, Korea University
[2)]Dept. of Electronics and Computer Engineering, Korea University
hklee@ispl.korea.ac.kr, hsko@korea.ac.kr

Abstract

This paper presents an efficient speech interactive agent rendering smooth car navigation on artificial reality and telexistence by employing embedded automatic speech recognition and embedded text-to-speech modules, all while enabling safe driving. A speech interactive agent is essentially a conversational tool providing command and control functions to users such as enabling navigation task and a variety of task manipulation through natural voice interactions between user and interface. To cope with the multiple random inputs received from external command buttons and events occurred by service applications on car navigation system, this provides resource negotiation rules using priority control based on inter-process communication, speech interactive helper function, multi-thread process and exception handling. The proposed system is tested and optimized on real car environments.

Key words: Speech interactive agent, embedded ASR, embedded TTS

1. Introduction

As the quality of computer science and technology is steadily improving, the realization of our dream projecting ourselves using robots, computers, and a cybernetic human interface becomes possible. Artificial reality and telexistence technology can be used to support some works in a variety of situations [1]. This includes efficient tools to facilitate control center design, to plan tasks in hazardous environments and to train works. The concept of artificial reality and telexistence expands to include projection in a remote real world or telexisting in a computer-generated virtual environment.

As the one of these fields, interface research to virtual environments (VE) has shown that VE interfaces pose new challenges to human-computer interface design [2]. Characteristics identified as particular to VE user interfaces include the need for continuous response and feedback in real time and support for various types of movement for navigation and interaction tasks. Human and machine interfacing (HMI) can be made more accessible and convenient when a conversational agent [3] is applied in order to facilitate dynamic knowledge interaction. Conversation is one of the most important interactions that facilitate dynamic knowledge interaction. People can have a conversation with a conversational agent that can talk with people by using automatic speech recognition (ASR) and text-to-speech (TTS) as a combined unit.

The proposed intelligent speech interactive agent is a software-based independent process that can be run in a virtual robot of a remote server-side or in a dedicated tele-operation system. The speech interactive agent recognizes the command, and then answers the information that is related with the user's requests. The speech interactive agent includes not only the embedded ASR and TTS to provide the prompt responses under the resource limited embedded system, but also the DSR to recognize the arrival place for autonomous navigation. This agent uses the communication protocol using the defined data format with other applications for command and control. Commanders have only to send the command code such as ASR-START or TTS-START to speech agent, and receive the ASR_RESULT or TTS-COMPLETE from speech agent. Thus, the software based speech agent plays a role in providing a transparent and easy interfacing to communicate each other. In addition, for extending the service quality and capability easily, human and machine interface provides a flexible, extensible and transparent form as a stand-alone process.

In this paper, the proposed agent is applied to a car environment in order to train a drive skill and navigate some objectives. To evaluate the performance of proposed speech interactive agent, real car test is performed. The content of this paper is as follows. The design concept of speech interactive agent using a combination of embedded ASR, and TTS is presented in Section 2. In Section 3, we describe the multi-modal human and machine communication for interacting between them. In Section 4, we conduct the experimental evaluation on a real car environment. Finally, in Section 5, we provide conclusive remarks.

2. Design of Speech Interactive Agent

2.1 Speech Interactive Agent

As a problem-solving paradigm, fusion process model using functional evaluation stage is employed [4].

Although car navigation system is deterministic, the use of multiple input sensors makes the system complex to cope with various situations. The proposed speech agent is decomposed into three separate processes; composition process of sensory sources, speech signal processing process and decision-making process. As shown in Figure 1, the composition process of sensory sources plays a role in combining input requests and guiding a next-step. Speech signal processing process provides a speech interaction means using speech recognition and text-to-speech functions. Decision-making process provides a role of user-friendly interfacing using a speech interaction helper function as well as self-diagnosis function using a speech interaction watch-dog module.

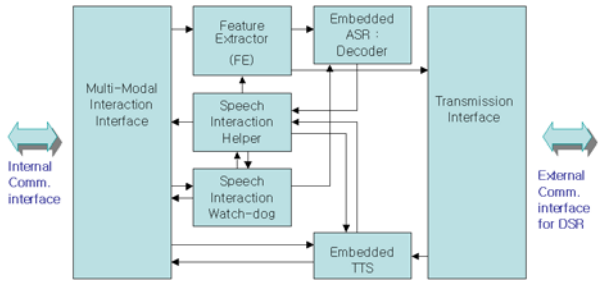

Fig. 1 Speech Interactive Agent (SIA) block diagram

The speech recognition system is classified into the embedded ASR and distributed speech recognition (DSR) system that is used via the wireless network using a CDMA 2000 terminal. Thus, the feature extractor based on ETSI v1.1.2 has the front-end role of passing the mel-cepstral features to eASR or DSR according to the scenarios without communicating between the speech agent and the application process. The eTTS utters the information when the event is requested by the user and application programs. The "speech interaction helper" provides the helper scenarios to the user when a recognition error occurs or an out-of-vocabulary is encountered. The "watch-dog" function monitors the service status of the eASR/eTTS to cope with the exception-handling event which can occur when a user pushes the external buttons during the service interval.

2.2 Sensory fusion

To perform the requests for speech interaction, firstly, sensory fusion model can be expressed by

$$Y_i = f(O/K, Y_{i-1}) \quad (1)$$

where i is a number of processing results, O is a observable sensory input, K is a domain knowledge, Y_{i-1} is status information being processing from previous time and $f()$ is the sensory fusion function to combine the sensory inputs and then control the current requests given the previous situation. The observable sensor input, O is expressed by

$$O = g_1(Mute)\cdot g_2(HF)\cdot g_3(R)\cdot g_4(Ptt)\cdot \prod_{i=0}^{k} g_5(E_i) \quad (2)$$

where M is a mute, HF is a hands-free, R is a remote controller, Ptt is a push-to-talk, E is a event occurred by service applications, and k is a number of applications being running simultaneously. Each input is independent each other as well as processed parallel. The variable, $g()$ is a function to observe and detect the sensor input. While a sensory input between g_1 and g_4 is a direct input from a sensor, g_5 is a transmitted input from application programs via the inter-process communication. The sensory inputs can be occurred simultaneously. However, the action to be performed promptly is always one function that is the most suitable in a given situation. This is due to the fact that hardware resource has limitation, and system can provides the robustness, consistency and efficiency in using a service. Thus, the fusing function, $f()$ should be considered with respect to the service quality and usability. In this paper, we apply the rule based decision function as a fusion function of respective inputs. In equation (1), K is a domain specific knowledge to provide combing rules as shown in Table 1. The given rule is decided by considering the service capability, priority and resource limitation, etc. Decision categories are composed of five decision rules.

Table 1. The negotiation rule table according to the priority control message

Current State / Previous State	eASR is requested	Application TTS is requested	CNS TTS is requested	Hands-Free Button pushed	Mute Button pushed
Hands-Free button enable	Disabled	Disabled	Enabled	Not applicable	Not applicable
Mute button enable	Enabled	Disabled	Enabled	Not applicable	Not applicable
eASR running (PTT button is pushed)	Previous eASR exits and new eASR runs	Previous eASR exits and eTTS starts	eASR runs continuously and CNS TTS starts	eASR exits	eASR exits
Application eTTS running	Previous eTTS stops and eASR runs	Previous eTTS stops and new eTTS starts	Application eTTS pauses and CNS TTS starts	eTTS stops	eTTS stops
CNS eTTS running	CNS eTTS starts and eASR runs	Previous CNS eTTS finishs and then application eTTS starts	Previous CNS eTTS stops and new CNS eTTS starts	Don't care	Don't care

2.3 Data fusion for speech interaction

When given the sensory fusion result, speech agent can decide the action to be performed. Next, data fusion model for speech interaction can be expressed by

$$Z = H_i(O_i)\cdot I(P)\cdot \mathrm{J}(\mathrm{Y}), \quad i = 1,...,3 \quad (3)$$

$$H_i(O_i) = h_i(O_i/M_i), \quad i = 1,...,3 \quad (4)$$

where i is a number of speech interaction tool and $H_i(O_i)$ is a speech interaction tool; 1)embedded speech recognition, 2)distributed speech recognition and 3)text-to-speech. Thus, the variable, O_1 and O_2 are speech sampling data and O_3 is a text data. Thus, $H_i(O_i)$ is decomposed as follows.

$$H_1(O_1) = h_1(O_1/M_1) \cong W_k = \arg\max_j L(O/W_j) \quad (5)$$

where $h_1(O_1)$ is a pattern recognizer using the maximum a posteriori (MAP) decision rule to find the most likely

sequence of words.

$$H_2(O_2) = h_2(O_2 / M_2) = h_2(O_2) \quad (6)$$

where $h_2(O_2)$ is a front-end feature extractor to pass the speech features into the back-end distributed speech recognition server.

$$H_3(O_3) = h_3(O_3 / M_3) \quad (7)$$

where $h_3(O_3)$ is a speech synthesizer function to read the sentences.

$J(Y)$ is a selecting function to choose a speech interaction tool. The currently selected speech module is just enabled. The variable, M_i is a given specific domain knowledge. M_1 is a acoustic model to recognize the word, M_2 is not used and M_3 is TTS DB. The variable, P is a procedural knowledge to provide a user-friendly service such as helper function. $I(P)$ is a function to guide the service scenario according to the result of speech interaction tool.

As a result, Z is an action flow to be performed sequentially. The final decision-making, $Z(t)$ represents the user's history to be processed when the decision is stored during a long-time period. This can provide the statistical information that the user frequently utilizes the specific function.

3. Multimodal Human/Machine Communication

Human can communicate with applications through a fusion agent by pushing an external buttons such as the touch screen, push-to-talk button, mute button and hands-free button. The fusion agent interprets and determines the priority level of multi-modal sources by integrating the inputs from multiple sensory modalities. The integrated command request [5] is passed to the speech interactive agent in order to provide the speech interaction with humans as shown in Figure 2.

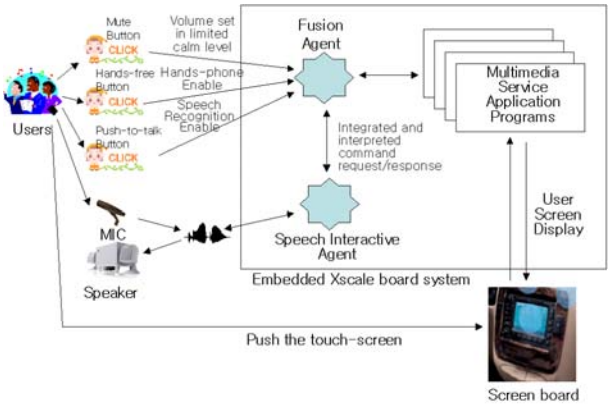

Fig 2. The block-diagram of the proposed system

3.1 Interface Modality

The system can provide the interface modalities such as push-to-talk button, touch screen, remote controller, mute button and hands-free button. The interface modalities have its own priority level and compensate information. Thus, the fused sensory control is required to extract final decision of user request.

3.2 Fusion Agent

As shown in Figure 2, the fusion agent integrates all button events occurring from interface modalities, as well as application events occurring from multimedia service applications. To do this, the fusion agent creates the multi-threads to provide asynchronous communications. After interpreting the requests, the fusion agent communicates with the speech interactive agent so that the fusion agent can provide the interpreted command. The challenging issue to be considered is that the speech resource for input/output devices is limited. Therefore, the fusion agent remembers the invoking count of used speech devices, and then negotiates with a speech interactive agent for resource usage.

3.3 Internal/External Communication

To communicate with each application programs connected with distributed network server, the system has communication capability using TCP/IP protocol. Meanwhile, to communicate with internal process within a virtual robot, the system uses inter-process communication (IPC) protocol between embedded multimedia service applications, fusion agent and SIA.

The transmission includes handling and relaying the request/response messages to be processed, and negotiation message to pipeline the permission for the use of the speech resource under negotiated rule-based scenarios. The data format is delineated in Figure 3.

Header part / Data part

STX	Application code	Message type	Scenario code	Priority level	Data type	Data Length	Data	ETX

Fields	Bits	Type	Remarks
Application code	8	_int8	Application code number N=1,....L : service application code number.
Message type	8	_int8	Send/Receive message type. (Common : 00-20, eASR : 21-40, DSR : 41-60, eTTS : 61-80)
Scenario code	8	_int8	The index number of service main screen and the index number of word lists for speech recognition.
Priority level	8	_int8	Message priority for resource negotiation.
Data type	8	_int8	Data type to be sent.
Data Length	8	_int8	Total data length to be sent.
Data	Variable	String	A message content to be sent (variable length). – eTTS utterance, news, traffic information and word lists for speech recognition etc.

Fig 3. Data Format for sending and receiving

4. Experimental Evaluation

For experimental evaluation, Multimedia Service applications, fusion agent and SIA together are implemented and tested on X-scale 400 Mhz, WindowsCE.NET AutoPC system as shown in Figure 4. In addition, proposed agent based system is applied to the car navigation environment.

4.1 Usability Issues

To provide the efficient tool for speech interaction and improve the performance of speech agent, usability issues are considered. These issues include as follows.

1) *Speech recognition start button*: by pushing the external button to talk.

2) *Disabling of speech recognition*: If the user dose

not speak any word for a while, the speech-waiting mode is closed automatically.

3) *Verification function*: TTS notify the recognition result.

4) *Undo function*: feedback to the previous state by pushing the external button if the recognition result is failed.

5) *Command mode*: have a global and local command. The user can choose the command mode; expand mode and local mode. Expand mode includes a global and local commands. Local mode includes only a local command.

6) *Out-Of-Vocabulary (OOV) rejection*: reject the word if there is no one in a given recognition list [7].

7) S*peech guidance*: TTS notify the guideline information for the use of service.

Fig 4. Data Format for sending and receiving

4.2 Speech Interaction Tools

The 11Khz, 16bit speech sampling rate for input and output is used for speech interactive agent. Each application programs communicate using the IPC protocol respectively according to the negotiated protocols.

The speech agent has one eASR [6][8][9] thread and one DSR front-end thread. Each thread uses the shared memory and Mutex to synchronize the objects. The total number of recognizable words is more than 5,000 words. However, the tree-based dynamic word recognition approach is applied according to the operational scenarios on the CNS domain. In addition, the speech agent has two eTTS [6] threads. One channel for eTTS output is related with the CNS, and the other is related with application services. The eTTS output related with the CNS has higher priority than others. Thus, the eTTS output related with the CNS never stops or pauses.

Driving test is performed on a real car as delineated in Table 2. The driving speed was done at a low speed between 20 and 60 Km/H while high speed was between 70 and 110 Km/H. A total of 40 men and women are tested on a Hyundai EF-Sonata and Samsung SM5 car respectively. The number of recognizable words is 100 on the given scenario respectively. In addition, the execution time and code sizes of the eTTS are also optimized as in Table 3.

Table 2. Test results on a car

	office	low-speed	high-speed	average	Car
Off-line	99.69	94.44%	92.10%	-	Avante (1800CC)
Men	-	95.4%	96%	95.7%	EF Sonata, SM5(2000CC)
Women	-	89.5%	89.2%	89.3%	
Average	-	92.5%	92.6%	92.5%	

Table 3. The eTTS average speed (11Khz, 40M DB)

Input Text (Bytes)	Output Sound (Bytes)	Tri-phone num.	Response time (milliseconds)			
			Language Processing	Speech Synthesize I	Speech Synthesize II	Total
51	135916	48	142	260	1177	1579
95	255196	92	332	489	2348	3169
111	270756	98	232	558	2225	3015
152	449662	150	391	727	3980	5098
222	531512	196	454	1096	4023	5573

5. Conclusion

In this paper, we proposed an efficient speech interactive agent rendering smooth car navigation by employing speech interaction tools on a car navigation environment. Experimental evaluation showed and confirmed that speech interaction tool can provide the efficient method for human and machine interface even in the field of artificial reality and telexistence.

References

1. K. Oyama et al,. “Experimental Study on Remote Manipulation Using virtual Reality,” Presence, Vol. 2, No. 2, 1993, pp.112-124.
2. Kaur K, Sutcliffe A, Maiden N, “Improving interaction with virtual environments,” In:Leevers DFA, Benest ID(eds), The 3D interface for the information worker, IEEE, London.
3. Aakay, M., Marsic, I., Medl, A., Guangming Bu, "A system for medical consultation and education using multimodal human/machine communication," IEEE Trans-Information Technology in Biomedicine, Vol 2 , Issue: 4 , Dec. 1998.
4. Richard T. Antony, ***Principles of Data Fusion Automation,*** Artech house, 1995.
5. Gerard J. Holzmann, ***Design and Validation of Computer protocols***, Prentice Hall, 1991.
6. X. Huang, A. Acero and H. Hon, ***Spoken Language Processing***, Prentice Hall PTR, 2001.
7. Taeyoon Kim and Hanseok Ko, "Uttrance Verification Under Distributed Detection and Fusion Framework", Eurospeech 2003, pp. 889~892, Sep, 2003.
8. Jounghoon Beh and Hanseok Ko, "A Novel Spectral Subtraction Scheme For Robust Speech Recognition: Spectral Subtraction using Spectral Harmonics of Speech," ICME 2003, III 633 ~ III 636, Jul, 2003.
9. Wooil Kim, Sungjoo Ahn and Hanseok Ko, "Feature Compensation Scheme Based on Parallel Combined Mixture Model", Eurospeech 2003, pp. 677~680, Sep, 2003.