Avestia Publishing
Journal of Multimedia Theory and Application
Volume 1, Year 2014
Journal ISSN: 2368-5956
DOI: 10.11159/jmta.2014.005

A Framework for Audiovisual Dialog-based Human Computer Interaction

Georgios Galatas[1, 2], Fillia Makedon[1]
[1]The University of Texas at Arlington, Department of Computer Science
500 UTA Boulevard, Arlington, USA
[2]NCSR Demokritos
Athens, Greece
georgios.galatas@mavs.uta.edu; makedon@uta.edu

***Abstract** - In this paper, we describe a framework for audiovisual human computer interaction using a PC. The architecture of our system is based on recognized speech from both audio and visual information, and a dialog system that is capable of deriving the user's intentions, taking into account contextual information. The design of our system goes beyond the traditional paradigm of using vocal commands to control a computer which is usually application dependant. In order to do so it employs an adaptive module, able to select an appropriate grammar that suits the program used at a particular time. Furthermore, our system utilizes the visual modality in addition to audio, for increased word accuracy. We carried out a number of experiments during which we acquired promising results, therefore showing the great prospect of our framework.*

Keywords: HCI, speech recognition, dialog systems, computer control.

1. Introduction

Human Computer Interaction (HCI) based on audiovisual inputs can benefit to users greatly. More importantly, the effects may be more prominent for users with disabilities, since traditional computer input methods can be proven to be ineffective. Tactile devices, such as the keyboard and mouse, are difficult to use in many circumstances and are generally not considered to be a natural form of interaction. Various approaches have been proposed and a number of implementations exist, that attempt to utilize other modalities for the task of controlling a computer. Since speech is one of the most natural means for human communication, its use is widespread in the literature. Kader et al. [1] created a system based on the MS API and Voice-XML for conducting basic actions of the operating system. This work was extended by Kadam et al. [2] by combining Hidden Markov Models (HMMs) with Dynamic Time Warping (DTW) in order to recognize spoken vocal commands. Sporka et al. [3] used speech in order to evaluate the effectiveness of spoken and other vocal information for controlling videogames. Various proposed system implementations also exist using either software or hardware [4, 5]. Furthermore, numerous applications are used in everyday life for dictation and interaction with devices such as mobile phones and cars [6]. Finally, other sources of information have been investigated for the task of controlling a computer, primarily employing visual information of the user's gaze, either with specialized hardware or cameras [7, 8, 9].

Existing solutions are built having in mind a specific OS or application and are bound to macros or static templates to perform more complicated actions. Additionally, they are generally limited due to the fact that most modern operating systems are designed primarily for interaction with a keyboard and mouse. Systems based on eye gaze may be able to provide a "mouse-like" interaction, but can be quite inefficient at actions involving clicking, scrolling, and fine positioning. Finally, speech based systems depend

Date Received: 2013-10-30
Date Accepted: 2014-03-25
Date Published: 2014-04-23

heavily on the accuracy and robustness of the underlining automatic speech recognition engine.

According to [14] more sophisticated interaction can be achieved by using natural input and output. This enables users to perform more quickly, more complex tasks, more accurately. In this paper, we present the framework of an audiovisual system for computer control which addresses the shortcomings of available systems by employing audiovisual dialog-based interaction. Our architecture collects not only audio signals but also visual articulation information from the speaker's face, in order to increase the word recognition accuracy and robustness. At the same time, rather than requiring cumbersome specialized hardware, it utilizes the low cost microphone and camera that most laptops are equipped with. Additionally, a dialog system (DS) is used to generate interaction events, such as clicking, scrolling, opening or closing programs, etc., based on verbal cues issued by the user, utilizing a grammar that is specific to the application used. It is the authors' belief that this architecture is able to provide natural speech-based interaction with existing operating systems without requiring a complete redesign of the GUI.

2. Methodology

Our system architecture is comprised of two main components, action generation, which interprets the intentions of the user into actions and the audiovisual speech recognition (AVSR) component, which acts as a front-end to the former.

2.1. Audiovisual Speech Recognition

The AVSR module that is integrated to the system is used for recognition of vocal commands by the user. This module utilizes the computer's built-in microphone and camera, in order to capture the audio signal of the speaker's voice as well as the mouth region of speaker's face (Figures 1a, b). This approach ensures higher reliability for the system, since each stream captures correlated information from a different source, which can complement each other.

The features extracted from the audio stream are the well-known Mel Frequency Cepstral Coefficients (MFCCs) and the features extracted from the video stream are the 2-D Discrete Cosine Transform (DCT) coefficients from the region of interest (ROI) of the speaker (mouth). The size of the ROI is 64x64 pixels and the features extracted for each frame are 45, corresponding to the highest energy coefficients. An example of the visual feature extraction is shown in Figure 1c.

The statistical modelling is carried out by means of HMMs and the implementation is based on HTK [10]. The feature vectors fed to HTK are modified in order to contain features from both streams, through concatenation. The Baum-Welch algorithm is used for training the models and the Viterbi algorithm is used for recognition. The recognized commands are finally used by the system in order to execute a specific action of the OS or the application used.

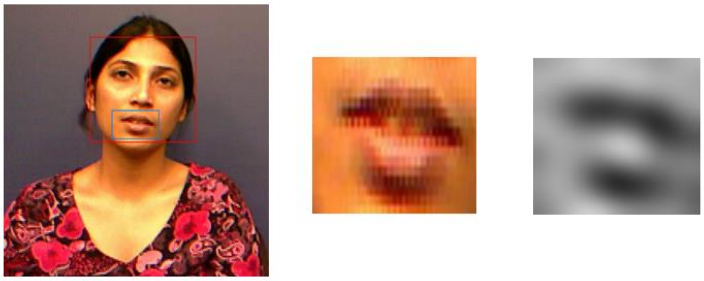

Figure 1. Face and mouth tracking (left) , Extracted mouth region (middle), Inverse DCT reconstructed mouth (right).

2.2. Action Generation

In order to understand the intentions of the user we developed a DS which is able to infer the user's intentions by taking into account vocal commands of the AVSR module, contextual information (the current OS "state") and the history of the interaction. Figure 2 depicts our DS's architecture, which is composed of the AVSR component [11], an NLU component from Olympus [12] and a dialog manager (DM). For the NLU we have created a grammar, in order to retrieve the meaning of the user's utterance (vocal command) which covers basic interaction with an OS. For specific applications, such as image manipulation, emails, writing text etc., we have created other grammars that work in conjunction with the more general one. The DM is responsible for integrating information from the NLU and OS and history, to identify regions of interest and regions where the user may interact with the system (such as an "OK" button). The system actions we defined for our DS essentially correspond to commands for the operating system. Available actions include click (left / right), move mouse, tab, escape, help, forward, backward and others. For specific applications we may have additional actions as well. The DS is called when the user provides a vocal cue, such as "click here" or "search for available apartments in the area".

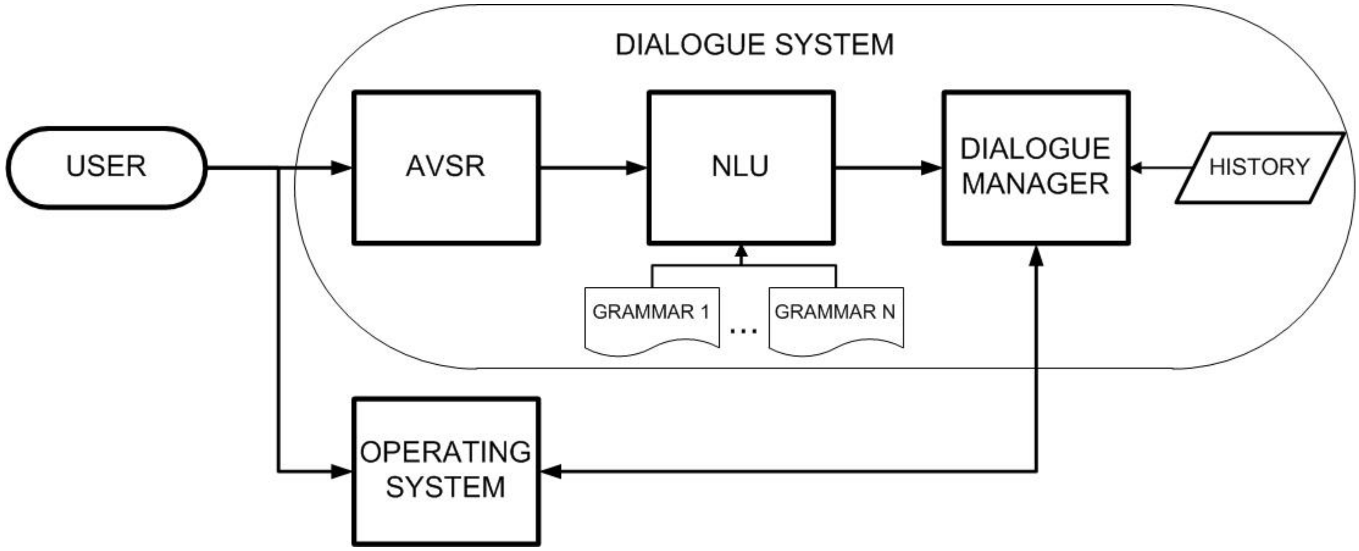

Figure 2. System overview. Our system can be easily configured to interact with various operating systems.

We opted to use a DS, as it is much easier to manage and much more scalable than using a set of complicated vocal commands, that would be specific to each application. To model the dialog for this application, we followed the Information Seeking [13] paradigm. The basic idea is that the system prompts the user (or waits) for some pieces of information, necessary to perform a specific task. When the system has enough information, it is able to perform the task. For example, a flight booking dialog system would prompt the user for dates, origin and destination, before performing the booking. In our case, the system does not reply to the user, but rather performs a task, such as to open an internet browser and use a search engine, or magnify a part of the screen. The DS also takes contextual information into account, so when the user wants to search for a keyword and is currently reading a document, the system will search within that document; when a list of files is in the foreground of the desktop, the system will look for a file whose name matches the keyword; and when the user is browsing the web, the system will search for the keyword using a search engine.

2.3. NLU Grammars

To account for the various meanings the user's utterance may have, depending on the application, and to promote simplicity by allowing phrases to map to several meanings, we have developed several grammars, one for each interaction scenario. Depending on the current focus of the operating system (the application being used), as well as contextual information (applications running in the background, etc.), we select the appropriate grammar. Currently, we have grammars for generic interaction, i.e. when no program is selected or when the user is browsing through the operating system's settings and folders/files. Another grammar is designed for internet browsing and another for text processors. In the future we plan to develop more advanced grammars, targeted for professional programs, such as architectural design, video and photograph editing, etc. Table 1 shows part of the internet browsing grammar.

Table 1. Part of the internet browsing grammar, where * denotes optional presence of that string.

[GoTo]	[Website]	[WinTab]	[Navigate]
(Forward)	(Address)	(Window)	(go to* the* GoTo)
(Back)	(Bookmark)	(Tab)	(go to* Website)
(NextTab)			(open Website in* new* WinTab*)
(PreviousTab)			

3. Results

We report preliminary results for the word recognition accuracy of the user's intended action. We conducted experiments on the accuracy achieved in comparison to a traditional audio-only speech based system in realistic scenarios under the presence of audio noise and various visual degradations. In addition we tested its performance under the influence of audio noise and partial absence of visual information caused by a black block (Figure 3a), affecting lip movements and causing slurred speech. Our experimental paradigm focuses on the recognition and successful execution of 11 tasks by our system. These 11 tasks are shown in Table 1 and focus on basic navigation functionality using an open-source web browser. Furthermore, they constitute a subset of the potential commands and actions that could be used and recognized by the system, but they were considered adequate for the "proof of concept" experiments that were conducted.

Our results are presented in figures 3b and c respectively, and show high accuracy for ideal conditions (>95%) and a significant gain in robustness in comparison to the baseline audio-only system in a variety of noise levels, even when both the audio and visual information are affected.

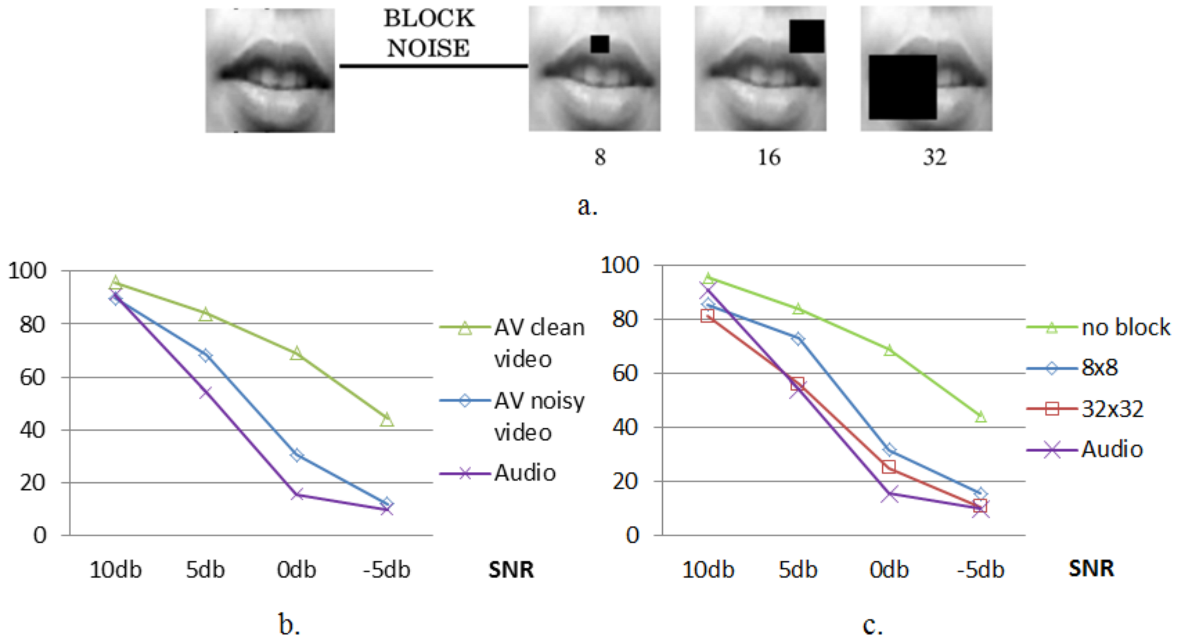

Figure 3. Block noise example (a.), recognition accuracy results for noisy AV (b.) and noisy audio anblock noise (c.)

4. Conclusions

Our system architecture combines audiovisual speech recognition and a dialog manager in order to effectively derive the user's intentions. Such a framework simplifies HCI and additionally enables the disabled to be able to interact with computers thus enhancing their quality of life and increasing their productivity. Our experimental results are promising, attaining high recognition accuracy for the user's intentions. This claim was verified even under adverse audio and visual conditions, simulating a variety of real life scenarios.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grants No. NSF-CNS 1035913, NSF-CNS 0923494 and NSF-IIS 1258500. The authors would like to thank professor G. Potamianos and Alexandros Papangelis for their expertise.

References

[1] M.A. Kader, B. Singha, M.N. Islam "Speech enabled operating system control",in 11th International Conference on Computer and Information Technology, 2008, 448-452.

[2] J.A. Kadam, P. Deshmukh, A. Kamat, N. Joshi, R. Doshi "Speech oriented computer system handling", in International Conference on Intelligent Computational Systems (ICICS'2012), 2012, 7-10.

[3] A.J. Sporka, S.H. Kurniawan, M. Mahmud, P. Slavik "Non-speech input and speech recognition for

real-time control of computer games" ASSETS'06, Proceedings of the 8th International ACM SIGAACCESS Conference on Computers and Accessibility, 2006, 213-220.

[4] R. Grant, P.E. McGregor "Method for integrating computer processes with an interface controlled by voice actuated grammars", US patent no. 6208972, 2001.

[5] M.H. Van Kleeck, S.S. Hysom "Voice-controlled computer simultaneously displaying application menu and list of available commands" US patent no. 5890122, 1999.

[6] X. Huang and L. Deng "An overview of modern speech recognition" Handbook of Natural Language Processing, 2010, 339-366.

[7] V. Pasian, F. Corno, I. Signorile, L. Farinetti "The Impact of Gaze Controlled Technology on Quality of Life. Gaze Interaction and Applications of Eye Tracking: Advances in Assistive Technologies" 2012, 48–54.

[8] EyeWriter, [Online] Available at: http://www.eyewriter.org/ consulted 27 Apr. 2013.

[9] J. Magee, Z. Wu, H. Chennamaneni, S. Epstein, D.H. Theriault, M. Betke "Towards a multi-camera mouse-replacement interface" in Proceedings of Patter Recognition in Information Systems (PRIS'10), 2010, 33-42.

[10] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, P. Woodland "The HTK book", 2002, Tech Rep.

[11] G. Galatas, G. Potamianos, F. Makedon "Audio-visual speech recognition incorporating facial depth information captured by the Kinect" 2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO), 2012, 2714-2717.

[12] D. Bohus, A.I. Rudnicky "The RavenClaw dialog management framework: Architecture and systems", Computer Speech & Language, 23(2), 2009, 332-361.

[13] V. Rieser, O. Lemon "Reinforcement Learning for Adaptive Dialogue Systems, A data-driven methodology for dialogue management and natural language generation" Series: Theory and Applications of Natural Language Processing, 2011, XVI, 256p.

[14] A. Jaimes, S. Nicu "Multimodal human–computer interaction: A survey, Computer vision and image understanding" 108(1), 2007, pages 116-134.