BLENDING SPEECH AND VISUAL INPUT IN MULTIMODAL DIALOGUE SYSTEMS

Manolis Perakakis, Michail Toutoudakis and Alexandros Potamianos

Dept. of Electronics and Computer Engineering, Technical University of Crete, Chania 73100, Greece

{perak, mixtou, potam}@telecom.tuc.gr

ABSTRACT

In this paper the efficiency and usage patterns of input modes in multimodal dialogue systems is investigated for desktop and personal digital assistant (PDA) working environments. For this purpose a form-filling travel reservation system is designed and implemented that efficiently combines the speech and visual modalities; three multimodal modes of interaction are implemented, namely: “Click-To-Talk”, “Open-Mike” and “Modality-Selection”. The three multimodal systems are evaluated and compared with the “GUI-Only” and “Speech-Only” unimodal systems. User interface evaluation includes both objective and subjective metrics and shows that all three multimodal systems outperform the unimodal systems on the PDA environment. For the desktop environment the multimodal systems score better than the “Speech-Only” system but worse than the “GUI-Only” system. In all evaluation experiments, the synergy between the visual and speech modality was significant: the multimodal interface was better than the sum of its (unimodal) parts. Results also show that users tend to use the most efficient input mode.

Index Terms— Speech communication, Graphical user interfaces, Natural language interfaces, User modeling, Speech recognition

1. INTRODUCTION

The emergence of powerful mobile devices, such as personal digital assistants (PDAs) and smart-phones, raises new design challenges and constraints that could be better addressed by a combination of more that one modalities. Few guidelines exist for selecting the appropriate mix of modalities [1]. It is established that visual modality is more efficient than speech [2], while speech is a more natural interaction mode. However, it is often the case when designing multimodal user interfaces, that the developer is biased either toward the voice, or the visual modalities. This is especially true, if the developer is voice-enabling an existing graphical user interface(GUI)-based application or building a GUI for an existing voice-only service. Our goal is to follow an approach that respects both modalities, by creating an interface that is both *natural* and *efficient*. To do so we also need to exploit the synergies that exist between the various interaction modes. By examining the relation between user satisfaction, user behavior (input modes usage) and objective metrics, the interface designer can decide which mode is the best (most efficient and user satisfactory) at each point in the interaction.

We have implemented and evaluated a travel reservation form-filling multimodal dialogue system, for both desktop and PDA environments. The desktop system combines keyboard, mouse and speech input while the PDA system combines pen and speech input. Three multimodal modes were implemented, namely: “Click-To-Talk”, “Open-Mike” and “Modality-Selection”. For “Click-To-Talk” interaction the visual modality is the default input mode, while for “Open-Mike” interaction, speech input is the default mode. “Modality-Selection” is a mixture of “Click-To-Talk” and “Open-Mike” interaction. The three multimodal systems are evaluated and compared with the unimodal systems (“Speech-Only”, “GUI-Only”). Our aim is to investigate user usage of input modes, as well as the relative efficiency of each interaction mode and system.

2. UNIMODAL AND MULTIMODAL INTERACTION

Our multimodal dialogue system is a travel reservation system (flight, hotel and car reservation) that extends the Bell Labs Communicator described in [3, 4]. The user can communicate with the system using speech and/or GUI. Overall, five different interaction modes were implemented; two unimodal ones, namely, “GUI-Only” and “Speech-Only” and three multimodal ones, namely, “Click-To-Talk”, “Open-Mike” and “Modality-Selection”. The various interfaces and interaction modes are presented next.

2.1. Unimodal GUI Interaction

The application GUI is generated automatically from the application ontology and the interface specification as described in [5]. It depicts the application state, using a series of forms; each form contains attribute-value pairs, each employing label and text-field/combo-box components, respectively. Two versions of the GUI are implemented: a desktop version which allows for keyboard and mouse input (GUI uses both text fields and combo boxes - see [6]) and a PDA version which only allows for pen input (GUI uses only combo boxes - see Fig. 2).

Selected attribute fields, e.g., “departure time”, “airline” and “car rental company” are implemented as a combo box in the desktop GUI. Only attribute fields that have less than ten value options were implemented as combo boxes in the desktop GUI. For the PDA GUI *all* data entry fields are implemented as combo boxes due to the slow text input methods available on such devices. The number of options available to the user in some of these combo boxes is quite large, e.g., 250 choices for the “hotelname” attribute.

The following features are common for both the desktop and PDA GUI: (1) ambiguity is shown as a pull-down box with a list of choices and highlighted in red, (2) error messages as represented in the GUI as pop-up windows, (3) fields and buttons that become inaccessible in the course of the interaction are “grayed out”, and (4) the context (or focus) of the interaction is highlighted.

2.2. Unimodal Speech Interaction

The original Communicator uses the BLSTIP [7] telephony platform. To further develop and explore multi-modality features on the Communicator, a simple yet flexible audio platform was designed

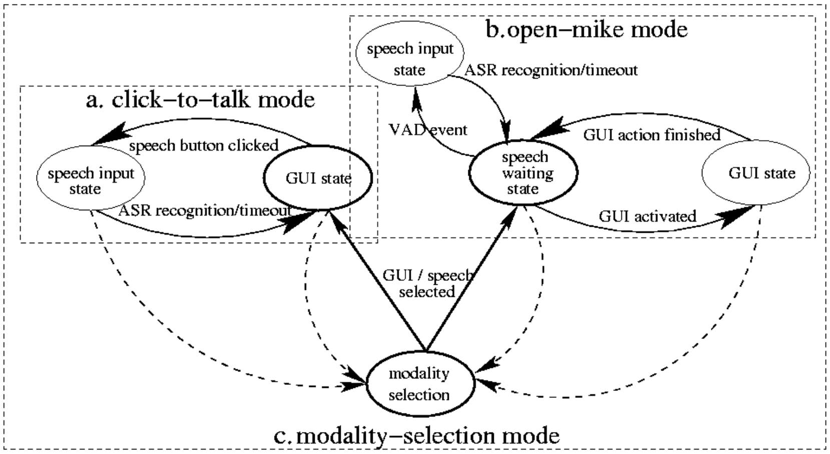


Fig. 1. State diagrams of the three multimodal modes : "Click-To-Talk", "Open-Mike" and "Modality-Selection"

and implemented. The audio platform interfaces with Bell Labs recognizer [7] and the FreeTTS [8] synthesizer through network sockets. The audio platform can be run on both desktop computers and mobile devices (for various OS). The audio platform implements *Voice Activity Detection* (VAD) and *barge-in*, i.e., users speaking over system prompts. The detailed description of the platform is beyond the scope of this paper.

The "Speech-Only" interface is identical to the one described in [5, 3, 4]. In brief, the spoken dialogue manager promotes mixed-initiative system-user interaction. All types of user requests and user input are allowed at any point in the dialogue, i.e., the full application grammar is active throughout the interaction. The system prompts are focused and try to elicit specific information from the user, e.g., the value of an attribute. Explicit confirmation is used only to confirm the values of the attribute at the form level, e.g., for all flight leg user supplied information. Implicit confirmation is used in all other cases throughout the interaction The main difference in the speech interface employed for the multimodal interaction modes, is in speech output: the speech prompts are significantly shortened for all three multimodal interaction modes.

2.3. Multimodal Interaction

Three different multimodal (MM) interaction modes have been implemented for combining the visual and speech modalities. The output interface is common for each interaction mode to allow us to better investigate the effectiveness of the "optimum" input modality mix. The visual output is identical to the corresponding "GUI-Only" mode. On the other hand, audio output prompts were significantly shortened compared with the unimodal "Speech-Only" case. In general, speech output was mainly used as a way to grab the attention of the user, emphasizing information already appearing on the screen. The speech interface was identical for all three multimodal modes.

Note that in all three multimodal modes only one modality is active at a time, i.e., the system does not allow for concurrent multimodal input[1]. In our current multimodal implementation, visual input is not allowed (GUI is "grayed-out") while speech input is active.

The state diagrams of the three multimodal modes are shown in Fig.1 (for a more detailed description see [6]). "Click-To-Talk" mode assumes that visual input is the default input modality and allows users to switch to the speech modality by clicking on a speech activation GUI button. "Open-Mike" mode assumes that speech is the default input modality and allows the user to switch to visual input by clicking on the GUI. "Modality-Selection" mode is a mixture of "Click-To-Talk" and "Open-Mike" modes. "Modality-Selection" attempts to better balance the visual or speech input modalities and correct the bias toward one or the other modality often found in today's multimodal systems. It is a simple version of the adaptive modality tracking algorithm proposed in [4].

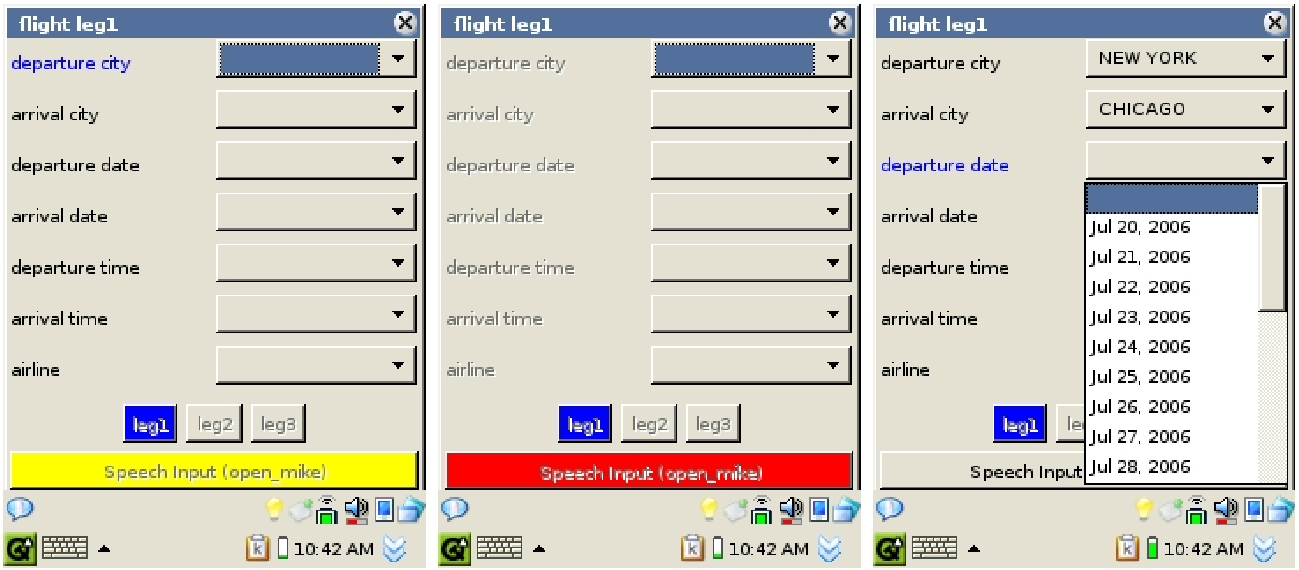
For the "Modality-Selection" mode, the input modality is selected in a static way. The system selects the "optimal" input modality (speech or visual, thus transitioning to the default state of the "Click-To-Talk"/"Open-Mike" respectively) at each interaction turn. For the desktop application, visual input is selected when a *combo box* is available for the attribute field that is in focus (expected user input), otherwise the speech modality is selected. For the PDA application, visual input is selected if the *combo box* that is in focus contains fewer than 25 values, otherwise the speech modality is selected.

In Fig. 2, examples from the "Modality-Selection" mode running on the PDA, are shown. Initially the interaction focus is on "departure city", the speech modality is selected (over 25 options available) and the system goes to "speech waiting" state. User input "from New York to Chicago" activates the speech recognizer (VAD event) and the GUI becomes disabled ("speech input" state). Once the recognizer returns the recognized utterance, the GUI is updated and the modality is selected for the next turn ("modality selection" state). For the next turn, visual input is selected (focus is on "departure date" for which a combo box with less than 25 choices is available) and the system goes to the "GUI input" state. "Modality-Selection" mode offers to the user better control of the mix of the visual and speech modalities in the multimodal dialogue system.

3. EVALUATION

For the desktop application the two unimodal ("Speech-Only", "GUI-Only") and three multimodal systems were evaluated on five travel reservation scenarios of varying complexity: one/two/three-legged flight reservations, round trip flight with hotel/car reservation. Evaluation took place in an office environment with all software (spoken dialogue system, speech platform, visual interface) running

[1] For information-seeking/form-filling multimodal applications this is not a major limitation.



departure city NEW YORK

arrival city CHICAGO

departure date

Fig. 2. “Modality-Selection” interaction mode examples shown for PDA application; switching between “Open-Mike” and “Click-To-Talk” interaction modes (user input: “From New York to Chicago”).

on the same host computer. Ten non-native English-speaking users evaluated the five systems on all five scenarios, a total of 25 runs per user. Users did not have prior experience using spoken dialogue systems. Systems were evaluated in random order.

For the PDA application the “GUI-Only” and three multimodal systems were evaluated on five travel reservation scenarios of varying complexity[2]. Evaluation took place in a “quiet” office environment with all the back-end software (spoken dialogue system, speech platform) running on a host desktop computer and the front-end (visual interface) running on a Zaurus Linux PDA device. Nine non-native English-speaking users evaluated the four systems (thus 20 runs per user); four out of the nine PDA evaluation users also participated in the desktop evaluation.

Initially, each user is given a short introductory document which explains the system functionality with emphasis on the modes to be evaluated. Then to famimiliarize the user with the system, each user is asked to complete a demo scenario using all different modes. After finishing the demo scenario, each user is asked to complete all five scenarios using all modes. Upon completion of each run the user is asked to evaluate the system by filling out a questionnaire (subjective evaluation). Upon completion of all runs, an exit interview is conducted (user feedback and overall system evaluation).

3.1. Objective Measures

Objective evaluation measures for desktop and PDA systems are shown in Table 1. For each interaction mode, task completion, percent (of number of turns) of usage of the speech and visual input modalities, task and turn duration statistics for both systems are shown. Note that duration statistics are computed only for completed scenarios.

For the desktop application in terms of task completion, all modes are equivalent (no statistical significant difference) with the exception of the “Speech-Only” mode which performs significantly worse. In terms of task and turn duration, the “GUI-Only” mode is the fastest, followed by the three multimodal modes (no significant difference among them) and the much slower “Speech-Only” mode. Note that the difference in task duration between the GUI and multimodal modes is due to the average turn duration not the number of turns.

For the PDA application the results are quite different than for the desktop applications. In terms of task completion, the “GUI-Only” and all three multimodal modes are equivalent. In terms of task and turn duration, all three multimodal modes outperform the “GUI-Only” mode on the PDA. This is mostly due to the reduction of “efficiency” of the PDA GUI compared to the desktop GUI. In the PDA “GUI-only” system, all attributes fields are implemented as combo-boxes, some with numerous values (e.g., the “hotelname” combo-box contains 250 values). The user has to navigate through these combo boxes using only a pointing device, significantly increasing the task duration of the PDA “GUI-only” system, relatively to the desktop one.

Comparing the three MM modes we can see a higher speech usage (in terms of percentage of speech turns) in the “Open-Mike” mode compared to the “Click-To-Talk” and “Modality-Selection” mode for both desktop and PDA environments. This shows that users input mode usage is affected by the multimodal interface design. By comparing the two applications environments (desktop vs PDA) in terms of percentage of speech and GUI usage, it is clear that the speech usage is higher for the PDA environment for all three multimodal interaction modes. This increase is in the order of 10% absolute in percentage of speech turns. We conclude that as the relative “efficiency” of one mode increases[3], so does the usage of that

[2] The PDA evaluation scenarios were shorter (fewer forms had to be filled) than the desktop evaluation scenarios but otherwise identical. Results for desktop experiments are normalized for this effect; thus results for desktop and PDA are directly comparable.

[3] For PDA, the task duration for the visual input mode increases compared to Desktop and thus the relative efficiency of the speech input mode also increases.

	Objective metrics					Subjective overall
System/Metric	Task comple-	Speech/GUI	Avg. duration (sec)		Avg. # of turns	User satisfaction
	tion(%)	turns(%)	task	turn	per task	mean (std)
Speech-Only	62	100 / 0	145.06	11.9	12.23	3.56 (1.60)
Desktop evaluation						
GUI-Only	100	0 / 100	64.65	6.4	10.13	4.48 (0.83)
Click-to-talk	98	56 / 44	84.38	8.3	10.20	3.81 (1.13)
Open-mike	96	65 / 35	86.25	7.8	11.10	3.87 (1.30)
Modality-Selection	98	56 / 44	84.00	8.0	10.50	3.56 (1.60)
PDA evaluation						
GUI-Only	100	0 / 100	86.89	8.08	10.76	4.61 (0.57)
Click-to-talk	100	66 / 35	83.63	7.74	10.80	4.60 (0.53)
Open-mike	100	79 / 22	80.40	7.18	11.20	4.64 (0.54)
Modality-Selection	100	64 / 36	80.01	7.81	10.24	4.57 (0.63)

Table 1. Objective and Subjective Metrics for Desktop and PDA.

mode. Overall, input mode usage is affected by the mode efficiency; however, efficiency is not the only parameter affecting input mode selection by the user.

Next we compare the task duration of the three multimodal systems and the two unimodal systems for both the PDA and desktop environments. It is clear that in terms of task and average turn duration synergies exists between the visual and speech modalities. For example, the task duration for the multimodal systems is less than the average of the task duration of the "Speech-Only" and "GUI-Only" systems (weighted by speech and visual input usage). This clearly shows that the multimodal interface (if appropriately designed) is more than the "sum of its parts": there is a gain in average task and turn duration measures by combining speech and visual modalities. It is also clear from these experiments that the user is able to select the most appropriate input modality in each interaction turn and take advantage of the synergy between the speech and visual modalities.

3.2. Subjective Measures

Subjective evaluation measures for desktop and PDA system are also shown in Table 1. The overall (for all five quesitions) mean and standard deviation of the Likert scores are shown. For the desktop environment "GUI-Only" significantly outperforms all other modes. Among the multimodal modes, "Open-Mike" and "Click-To-Talk" modes both outperform "Modality-Selection" in terms of subjective measures. Note that despite the fact that the "Speech-Only" mode was the least efficient, users satisfaction of this mode is very close to the three multimodal modes and especially the "Modality-Selection" one. For the PDA environment we see that there is no significant difference among the four modes; "GUI-Only", "Open-Mike", "Click-To-Talk", "Modality-Selection" all receive high marks that are almost identical.

4. CONCLUSIONS

In this paper, we implemented and evaluated two unimodal and three multimodal travel reservation systems on the desktop and PDA environments. Our evaluation experiments outlined some basic facts of multimodal dialogue system design: (1) Synergies between the speech and visual interaction modes exist in multimodal interfaces; the systematic modeling of these synergies requires further research. (2) When changing the relative efficiency of the input modes in multimodal interfaces, user input mode usage also changes. (3) It is not always true that a multimodal (speech and visual) interface is more efficient or preferable to the unimodal interface. The "best" interface is both a function of relative unimodal interfaces efficiency and user usage.

Future work will focus on evaluating the unimodal and multimodal systems for varying levels of task complexity and unimodal interface efficiency (e.g., different speech recognition error levels). Through these experiments multiple measurement points for mode usage, unimodal and multimodal interface efficiency will be obtained; these results will help us better understand the relationship between efficiency, user satisfaction and input mode usage. By incorporating this knowledge into the multimodal dialogue system design process we aim at building adaptive multimodal interfaces that are natural, efficient and outperform traditional unimodal interfaces.

Acknowledgments This work was partially supported by the EU-IST-FP6 MUSCLE network of excellence.

5. REFERENCES

[1] V. Bilici, E. Krahmer, S. teRiele, and R. Veldhuis, "Preferred modalities in dialogue systems," in *Proc. ICLSP*, Beijng, 2000.

[2] P. Cohen, M. Johnston, D. McGee, S. Oviatt J. Clow, and J. Smith, "The efficiency of multimodal interaction: A case study," in *Proc. ICSLP*, 1998.

[3] A. Potamianos, E. Ammicht, and H.-K. Kuo, "Dialogue management in the Bell Labs communicator system," in *Proc. ICLSP*, Beijng, 2000.

[4] A. Potamianos, E. Ammicht, and E. Fosler-Lussier, "Modality tracking in the multimodal Bell Labs Communicator," in *Proc. ASRU Workshop*, 2003.

[5] A. Potamianos, E. Fosler-Lussier, and E. Ammicht, "Information Seeking Spoken Dialogue Systems-Part II: Multimodal Dialogue," *to appear in IEEE Transactions on Multimedia*, 2006.

[6] M. Perakakis, M. Toudoudakis, and A. Potamianos, "Modality selection for multimodal dialogue systems," ICMI, 2005.

[7] Q. Zhou, A. Saad, and S. Abdou, "An enhanced BLSTIP dialogue research platform," in *Proc. ICSLP*, 2000.

[8] "FreeTTS," http://freetts.sourceforge.net/docs/.