Building a Practical Multimodal System with a Multimodal Fusion Module

Yong Sun[1,2], Yu (David) Shi[1], Fang Chen[1,2], and Vera Chung[2]

[1] National ICT Australia
Level 5, 13 Garden Street, Eveleigh NSW 2015 Australia
[2] School of Information Technologies, J12
The University of Sydney, NSW 2006 Australia
yong.sun@nicta.com.au

Abstract. A multimodal system is a system equipped with a multimodal interface through which a user can interact with the system by using his/her natural communication modalities, such as speech, gesture, eye gaze, etc. To understand a user's intension, multimodal input fusion, a critical component of a multimodal interface, integrates a user's multimodal inputs and finds the combined semantic interpretation of them. As powerful, yet affordable input and output technologies becoming available, such as speech recognition and eye tracking, it becomes possible to attach recognition technologies to existing applications with a multimodal input fusion module; therefore, a practical multimodal system can be built. This paper documents our experience about building a practical multimodal system with our multimodal input fusion technology. The pilot study has been conducted over the multimodal system. By outlining observations from the pilot study, the implications on multimodal interface design are laid out.

Keywords: Multimodal system design, practical multimodal system, multimodal input fusion.

1 Introduction

A Multimodal User Interface (MMUI) allows input and/or output to be conveyed over multiple modalities. Empirical data shows that an MMUI is easy for people to use because the interface is more natural and intuitive. It is preferred by users for many applications, such as map-based military and transport planning. Driven by powerful, yet affordable input and output technologies currently becoming available, such as speech recognition and eye tracking, a functional MMUI becomes more attainable than it used to be. To understand a user's intension conveyed through multiple modalities, Multimodal Input Fusion (MMIF) integrates multimodal inputs recognized by signal recognizers and derives combined semantic interpretations from them. With a flexible and portable MMIF technique that can adapt to other types of input modalities, it is possible to attach recognition technologies to existing systems and build a multimodal system. And the building process can be low-cost and time-saving. We propose an open MMIF approach termed PUMPP (Polynomial Unification-based Multimodal Parsing Processor) in [7] that can handle two parallel input strings. An approach termed THE HINGE has also

J.A. Jacko (Ed.): Human-Computer Interaction, Part II, HCII 2009, LNCS 5611, pp. 93–102, 2009.

been proposed to understand MMIF result and trigger a system action in [6]. In an attempt to develop a plug-and-play MMIF technique, and an economic and practical MMUI, a tentative multimodal system is built based on the MMIF techniques previously developed. Its recognition components are all off-the-shelf products and its application background is a pre-existing system. These pre-developed systems, which can be calibrated/adjusted individually and integrated with very few re-adjustments, enable us to concentrate on how to economically link these pre-developed systems together with our MMIF techniques both in cost and time. This paper reports how the tentative multimodal system is built and a pilot study over it. Implications found in the experience are also discussed.

2 The MMIF Technologies Applied in the System

Because all related MMIF technologies have been documented in the previous papers, an overview of them is outlined here.

PUMPP [7] is used for MMIF in the tentative multimodal system. It accepts multimodal inputs during a multimodal turn. A multimodal turn refers to an opportunity or segment that comes successively to a user to express his/her meaning to a system in multimodal human computer interaction. The multimodal inputs accepted by PUMPP are recognized by recognizers from multimodal signals a user utters. An atomic unit recognized by a signal recognizer is termed as a symbol. A complete set of symbols recognized from multimodal input signals during a multimodal turn consists of a multimodal utterance. Because the tentative system uses speech and eye gaze as input modalities, the inputs of PUMPP in multimodal system are speech symbols and gesture symbols which belong to a multimodal utterance as demonstrated in Fig. 1.

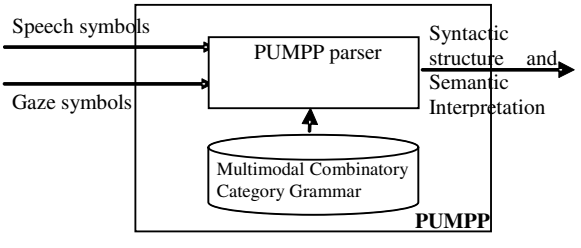

Fig. 1. The architecture of PUMPP

The internal of PUMPP parser has been introduced in [7], and it is not related to particular applications. Multimodal utterances that are legitimate in a multimodal interface are defined in multimodal combinatory category grammar; therefore, the multimodal combinatory category grammar is closely related to applications. In multimodal combinatory category grammar, most language information are condensed in lexical rules, therefore, this kind of grammar is mildly context-sensitive [8].

Although the semantic interpretation generated by PUMPP is represented in hybrid modal logic [4], which is a systematic semantic representation framework, the semantic interpretation still needs to be understood by a specific application. Intuitively, understanding MMIF result is mapping a semantic interpretation to a function of an

application. A mapping mechanism termed as THE HINDGE that accepts semantic interpretation in hybrid modal logic has been proposed in [6]. As shown in Fig. 2, THE HINGE is configured with a triggering condition to event table for an application. This table defines which triggering event should be generated under which condition. When PUMPP sends THE HINGE an MMIF result, THE HINGE finds a triggering condition compatible with the semantic interpretation in the MMIF result, and generates a triggering event corresponding to the triggering condition.

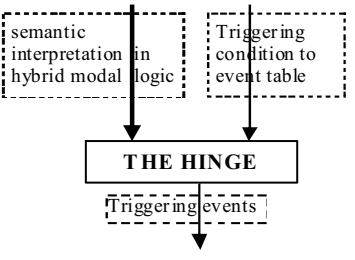

Fig. 2. The input and output of THE HINGE

3 The Tentative Multimodal System

3.1 Origin of the Multimodal System and Its Functions

There is an MPML3D (Multimodal Presentation Markup Language 3D) [5] player. It is a player that can deliver presentations described by scripts in MPML3D. In the presentations, life-like animated agents act in the role of virtual presenters that convey information with their multimodal expressiveness such as voice, gesture and body motion in a convincing and entertaining way.

Although the player has powerful multimodal presentation capability, it can only accept input from traditional keyboard and mouse. We attach speech recognition and eye tracking to it and try to enable its multimodal input capability.

The player running with a script is used as the background application. Based on it, a tentative multimodal system is built. In the tentative multimodal system, two animated agents introduce two MP3 players to a user, then, a user can ask questions regarding these two MP3 players with his/her speech and eye gaze. His/her eye gaze is used to select entities that he/she queries about.

3.2 Overview of the Multimodal System

Fig. 3 delineates the components in the tentative multimodal system and the relationships between them. A user's speech is caught by a Mic. and recognized by VoiceIn™ speech recognizer from Fonix [3]. The speech recognizer is running as a standalone thread. It is configured to recognize words and phrases defined as possible input in the tentative multimodal system.

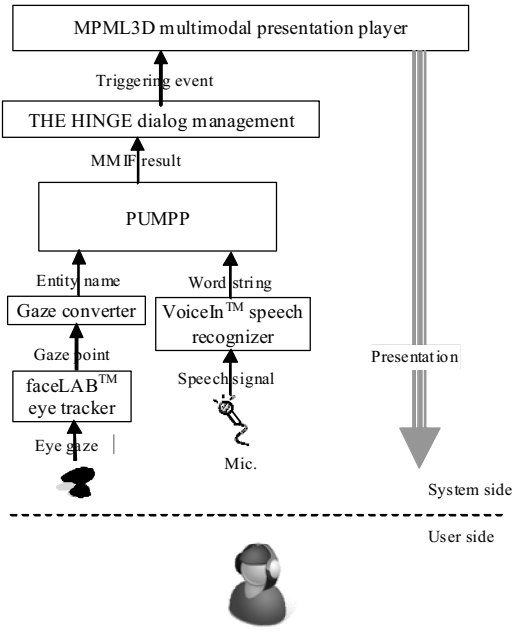

Fig. 3. The overview of the tentative multimodal system

A user's eye gaze is tracked by the faceLAB™ eye tracker, a commercial product from Seeing Machine [9]. The gaze converter accepts gaze points tracked by faceLAB™ eye tracker, deduces gaze points to the entity being gazed on, and provides the name of the entity as eye gaze input to PUMPP. The faceLAB™ eye tracker runs on an independent computer and sends tracked coordinates to the gaze converter. The eye gaze converter shares the location information of entities displayed on a screen in a presentation with the MPML3D player.

PUMPP, running as an independent thread, accepts a user's speech and eye gaze input and derives semantic interpretation of them. THE HINGE sends a triggering event to the MPML3D player according to the semantic interpretation received. To be compatible with input method of MPML3D player, keyboard and mouse, the triggering event is sent out as simulated combinations of key strokes.

Although the speech recognizer needs to be configured for an application and the eye tracker needs to be calibrated for each user, those are not very related to our MMIF techniques. We focus on what needs to be done to fit our MMIF techniques in a practical multimodal system in this paper.

3.3 Preparing a Multimodal Combinatory Category Grammar for PUMPP

Writing a multimodal grammar for PUMPP follows the guide in [2]. The following lexical rule of a PUMPP multimodal grammar illustrates the key concepts and tags.

```
<family closed="true" pos="PDet" name="PDet">
 <entry name="Primary">
  <complexcat>
   <atomcat type="np">
    <fs id="2">
     <feat val="3rd" attr="pers"/>
     <feat attr="index">
      <lf>
       <nomvar name="X"/>
      </lf>
     </feat>
    </fs>
   </atomcat>
   <slash mode="*" dir="/"/>
   <atomcat type="n">
    <fs id="2">
     <feat attr="num">
      <featvar name="NUM:num-vals"/>
     </feat>
     <feat attr="index">
      <lf>
       <nomvar name="X:appliance"/>
      </lf>
     </feat>
    </fs>
   </atomcat>
   <slash mode="^" dir="/"/>
   <atomcat type="n">
    <fs id="3">
     <feat attr="num">
      <featvar name="NUM:num-vals"/>
     </feat>
     <feat attr="index">
      <lf>
       <nomvar name="Y"/>
      </lf>
     </feat>
    </fs>
   </atomcat>
   <lf>
    <satop nomvar="X:sem-obj">
     <diamond mode="det">
      <nomvar name="P:proposition"/>
      <prop name="[*DEFAULT*]"/>
     </diamond>
     <diamond mode="ref">
      <nomvar name="Y"/>
     </diamond>
    </satop>
   </lf>
  </complexcat>
 </entry>
 <member stem="this"/>
</family>
```

The above fragment defines a category for a lexical family, and word “this” is one of its members. Each category has syntactic description and semantic description. The semantic description is encapsulated by “<lf> … </lf>”, and syntactic description is

above semantic description. The syntactic description says the functor category is "np(pers:3rd)/*n/^n". Because both the final "np" and the first "n" have the same feature ID "<fs id="2">", the final "np" can inherit the all attributes of the first "n". Semantic description is in hybrid modal logic. The semantic interpretation of final "np" is X, the semantic interpretation of first "n" is X as well and the second "n" has semantic "Y". The semantic description specifies that the semantic interpretation of the final "np" has property "Det", whose value is the stem of a member of this category family, and property "Ref" whose value is the semantic interpretation of the second "n". More information about hybrid modal logic can be found in [1]. In the multimodal grammar designed for the tentative multimodal system, there is no information to mark out which modality a symbol should be uttered from. This strategy makes the exchange of modalities possible because a symbol is legal to be uttered from either speech or eye gaze in the tentative multimodal system.

After all lexical rules of input symbols in a multimodal system are defined, MMIF results of a multimodal utterance can be determined by PUMPP. Fig. 4 illustrates an MMIF result. It contains the final category, syntactic information and semantic information.

The semantic information is represented in hybrid modal logic. A hybrid logic formula can be formed using both *logical and operator* "^" and the *satisfaction operator* "@". A formula @A(p) states that the formula *p* holds at the state *A*. For example, "@X0(<num> sg)" in Fig. 4 means property *num* is "sg" at state X0. A *nominal* is used to refer a state. The combination of these formulas fully describes semantic information. The semantic information in an MMIF result is regarded as the semantic interpretation of a multimodal utterance.

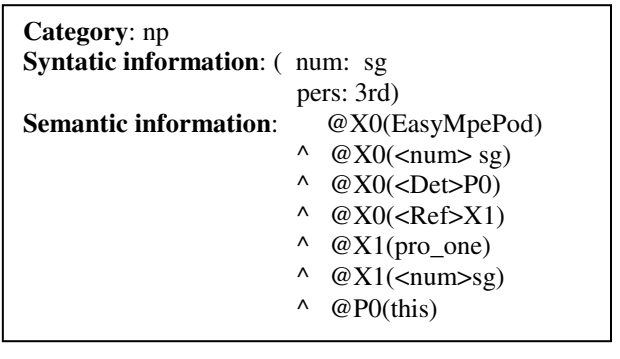

Fig. 4. The MMIF result of "this one" plus "EasyMP3Pod" selected by eye gaze

3.4 Configuring the Triggering Condition to Event Table for THE HINGE

After an MMIF result is derived by PUMPP, its semantic information is sent to THE HINGE module for dialog management. The multimodal utterances acceptable to the tentative multimodal system are in command and control style. Within the system, a user issues a multimodal command, and the system performs a function/action in responding to the command. Because multiple similar multimodal commands may result in the same system action, there is a multiple to one relation between

multimodal commands and a system function. To implement the multiple to one mapping between MMIF results and a system function, the common constituent of the semantic interpretations of these multiple similar commands are defined as the triggering condition for a system function.

```
    @E_1(how)
^   @E_1(<body>X_3)
^   @E_1(<mod>X_2)
^   @X_2(<mod>big)
^   @X_3(<pat>question)
^   @X_3(<Arg>X_7)
^   @X_5(EasyMP3Pod)
^   @X_7(storage)
^   @X_7(<mod>X_5)
```

Fig. 5. A triggering condition in THE HINGE

THE HINGE keeps a table listing pairs of triggering condition and the event to trigger the function. Fig. 5 illustrates a triggering condition. It is also represented in hybrid logic. Intuitively, a triggering condition separates a function in a system from others; and the collection of the separated functions should cover all functions in a system. This idea is illustrated in Fig. 6.

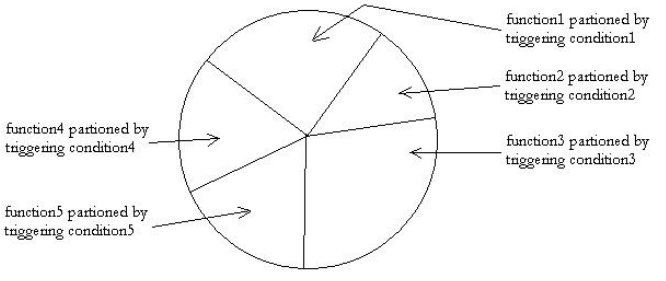

Fig. 6. A set of functions in a system partitioned by trigger conditions

4 A Pilot Study over the Tentative Multimodal System

4.1 Setup of the Pilot Study

In the pilot study, there is a virtual sales scenario where a team of two 3D animated agents present two MP3 players (EasyMP3Pod and MP3Advance) to a user. A user is seated in front of the monitor screen to communicate with the system, as shown in Fig. 7. All the components, except faceLAB™ eye tracker, of the system described in section 2 are running on a Dell Precision computer. The faceLAB™ eye tracker is running on an HP notebook computer. It communicates with Gaze converter through local network. As shown in Fig. 8, there are 6 entities in the screen for a user's eye gaze to select.

Fig. 7. The pilot study setup

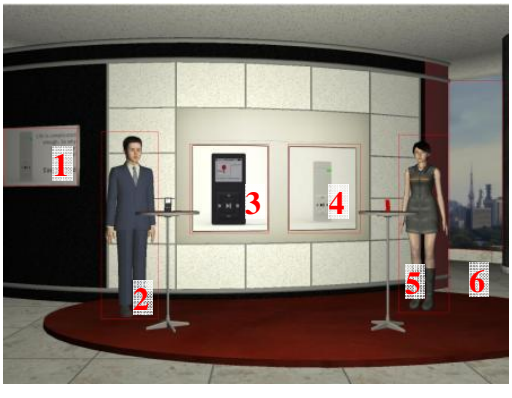

Fig. 8. Screen snapshot of the pilot study

Table 1. Multimodal utterances with speech and eye gaze as input modalities

#	Speech Input	Eye gaze fixation
1	How big is its storage	EasyMP3Pod
2	How big is its storage	MP3Advance
3	How many songs can it hold	EasyMP3Pod
4	How many songs can it hold	MP3Advance
5	How many songs can this powerful one hold	MP3Advance
6	How many songs can this advanced one hold	MP3Advance
7	Does this simple one have an FM tuner	EasyMP3Pod
8	What functions does this big one come with	MP3Advance
9	What is the storage medium of this easy one	EasyMP3Pod
10	What is the storage medium of this simple	EasyMP3Pod
11	Does this lovely one have a screen	EasyMP3Pod
12	Does this lovely one have a screen	MP3Advance
13	How many buttons does it have	EasyMP3Pod
14	How many buttons does it have	MP3Advance

To evaluate the performance of the tentative multimodal system, 10 subjects, 8 males and 2 females aging from 20 to 40, were recruited. Most of them are not native English speakers. They are asked to interact with the system with pre-scripted multimodal utterances about the two MP3 players, listed in Table 1. Each subject was asked to interact with the system with all multimodal utterances for 3 times. At the beginning of the study with a subject, the faceLAB^{TM} eye tracker is calibrated for the subject.

4.2 Results and Discussion

Out of 10 subjects, only the data from 9 was usable, since one user had difficulty comprehending the tasks, such that he could not achieve the goals of the task. Surprisingly, more than 90% utterances issued by a user were correctly understood by the system that was indicated by a correct response by the system. After a subject tried the multimodal system, he/she was asked to design two questions about the information in the presentation. We found that most of the questions could not be understood by the system.

These observations imply that a practical multimodal system is possible when a user's utterance is constrained within a certain scope. Otherwise, the user's utterances will not be understood by the system. Because any multimodal system is designed for a certain function or purpose, the multimodal utterances understandable to a multimodal system have to concentrate on a certain domain. When this fact is considered, a multimodal system can be re-defined as a system which a user can use utterances in a certain domain to communicate with. Facing a multimodal system, users often do not know which utterances are legitimate to it. As shown in Fig. 9, there are mismatches between the set of multimodal utterance understandable to a multimodal system and the set issued by users. There should be some facilities to constrain utterances users may issue, and make the utterances issued by a user fall in the multimodal utterance set understandable to a multimodal system.

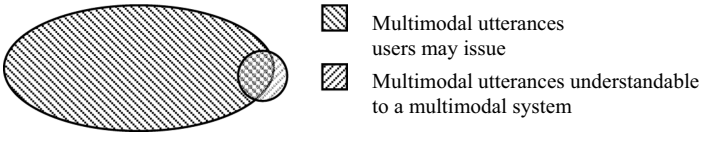

Fig. 9. Mismatches of two set of utterances in an MMUI

5 Conclusion and Future Work

This paper details the process building a practical multimodal system with our MMIF techniques. A practical multimodal system can be built by linking off-the-shelf recognition products and an existing application with PUMPP and THE HINGE. The works that need to be done to build the system have been enumerated.

Through the observation from the pilot study, we can conclude that a practical multimodal system with certain functions is attainable. The pilot study also reveals that the set of multimodal utterances a user may issue does not match or even far exceeds the set of multimodal utterances a multimodal system can support. Therefore, a

functional multimodal system should be able to constrain the multimodal utterances a user may issue. The closer these two sets overlap, the better the satisfaction index of a multimodal system will be. How to constrain a user's input in a multimodal interface can be a research issue.

References

1. Baldridge, J., Kruijff, M.G.: Coupling CCG and Hybrid Logic Dependency Semantics. In: 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia (July 2002)
2. Bozsahin, C., Kruijff, M.G., White, M.: Specifying Grammars for OpenCCG: A Rough Guide. The OpenCCG package (2006), http://openccg.sourceforge.net/
3. http://fonixspeech.com/
4. Kruijff. G. M.: A Categorial Modal Architecture of Informativity: Dependency Grammar Logic & Information Structure. Ph.D thesis, Charles University, Prague, Czech Republic (2001)
5. Nischt, M., Prendinger, H., Andre, E., Ishizuka, M.: MPML3D: a Reactive Framework for the Multimodal Presentation Markup Language. In: Gratch, J., Young, M., Aylett, R.S., Ballin, D., Olivier, P. (eds.) IVA 2006. LNCS, vol. 4133, pp. 218–229. Springer, Heidelberg (2006)
6. Sun, Y., Prendinger, H., Shi, Y., Chen, F., Chung, V., Ishizuka, M.: THE HINGE between Input and Output: Understanding the Multimodal Input Fusion Results In an Agent-Based Multimodal Presentation System. In: CHI 2008 extended abstracts on Human factors in computing systems, Florence, Italy, April 2008, pp. 3483–3488 (2008)
7. Sun, Y., Shi, Y., Chen, F., Chung, V.: An Efficient Unification-based Multimodal Language Processor in Multimodal Input Fusion. In: 19th Australasian conference on Computer-Human Interaction: Entertaining User Interfaces, Adelaide, Australia (November 2007)
8. Steedman, M.: The Syntactic Process. MIT Press, Cambridge (2000)
9. http://www.seeingmachines.com/