Designing Multimodal Conversational Interfaces Based on Empirical Studies of Human Communicative Behaviors

by

Yukiko I. Nakano

Submitted in Partial Fulfillment
of the Requirements
for the Degree
Doctor of Philosophy
(Information Science and Technology)

Graduate School of
Information Science and Technology

The University of Tokyo

2004

Acknowledgments

Along the way, during the long road toward completion of this thesis, I have met wonderful advisors, colleagues, and friends. Now, I would like to thank all those who have helped me along the way.

First, I would like to thank my thesis advisor Professor Toyoaki Nishida. He gave me a chance to continue my research at RISTEX, and kept encouraging me. He has also always given me innovative ideas concerning future human interfaces, and taught me how to formulate an argument. I also want to thank the other members of my thesis committee: Professors Mitsuru Ishizuka, Masaru Kitsuregawa, Shuichi Sakai, Sadao Kurohashi, and Nobuaki Minematsu. They used their precious time to review this thesis and give me insightful comments.

The first part of this thesis was supervised by Professor Tsuneaki Kato, while both of us were working at the NTT research laboratories. He patiently advised me, despite my lack of any computer science background. I also would like to thank all the people working with me at NTT: Takaaki Hasegawa, Hideharu Nakajima, Kenji Imamura, Hisashi Ohara, Masato Ishizaki, Tsuneko Nakazawa, and Masanobu Higashida.

I learned a lot about empirical-based research from Professor Justine Cassell at the MIT Media Lab. She provided guidance in analyzing human behaviors and gave me insightful and provocative discussions when I was writing my master thesis. Dr. Candy Sidner advised me with using her profound knowledge in computational linguistics, and masterly pointed out missing pieces in previous theories of discourse. Professor Deb Roy, meanwhile, a thesis reader of my master thesis, taught me the importance of originality in research.

I also want to thank all the members of Gesture and Narrative Language group (GNL) at the MIT Media Lab: Tim Bickmore, Hannes Vilhjálmsson, Kimiko Ryokai, Cati Vaucelle, and Dona Tversky. I would like to express special thanks to the Mack project members: Tom Stocky, Gabe Reinstein, and Ian Gouldstone. The model of face-to-face grounding would not have been successfully implemented without their efforts.

My colleagues at RISTEX helped me to complete this thesis: Toshihiro Mu-

rayama, Ken'ichi Matsumura, and Tomohiro Fukuhara. They helped me integrate and evaluate an agent system on a web-based application. I would also like to thank Professor Hideyuki Horii and Professor Hiroshi Komiyama, who have supported my research at RISTEX.

Members of the Nishida-Kurohashi lab at the University of Tokyo helped me analyze conversational data in Japanese. I would like to express special thanks to Daisuke Kawahara who helped me with his wonderful NLP techniques and Masashi Okamoto and Quing Li who persevered in analyzing gesture usage in Japanese.

I would also like to thank the researchers who gave me excellent comments. I would like to express my special thanks to David Traum, Lisa Harper, Michael Johnston, Barbara Tversky, and Matthew Stone for their stimulating discussions.

Finally, I would like to thank my family for supporting me at home and from a long distance.

Abstract

With the goal of rendering human-computer interaction more natural, this thesis addresses the design of Multimodal Conversational Interfaces (MCIs) based on analysis of human verbal and nonverbal communicative behaviors.

In order to design MCIs based on the empirical support of human communicative behaviors, we integrated the research methods used in multiple disciplines, such as communication science, linguistics, and media technology. Therefore, each study in this thesis consists of three steps: (1) collecting and analyzing real human verbal and nonverbal communicative behaviors; (2) establishing a model of the human communication protocol, based on the results of the empirical study; and (3) implementing a prototype system based on the model and evaluating the system through conducting a user study. By employing this approach, we propose designs for MCI components and build prototype systems showing how the components work.

First, based on the analysis of real spoken dialogues, the characteristics of the topic, the dialogue history, and the user's levels of understanding respectively were found to be the determinant factors for utterance content. In system implementation, these factors were used as utterance content selection rules within a content planning mechanism.

Subsequently, in order to handle multimodal human-computer communication, we extended the content planning mechanism and built a prototype system called MID-3D. This system keeps track of the user's viewpoint in a virtual world as the dialogue history, and interactively provides the user with instructions according to their individual perspectives (what the user(s) can see).

While dialogue state management is an essential aspect of human-computer interaction, we studied face-to-face grounding, where nonverbal behavior plays a critical role. Based on the analysis of direction-giving dialogues, a face-to-face grounding model was established and subsequently implemented into an embodied conversational agent, MACK.

To make a conversational agent capable of performing appropriate gestures, we analyzed human presentations to investigate the relationship between the linguistic information in an utterance and gesture occurrence. The results of the analysis are

used as gesture decision rules in CAST, which converts text to agent animations synchronized with speech. CAST was then integrated into a web-based automatic presentation generation system named SPOC.

Over the course of our evaluation experiments, we found some positive results supporting our empirical-based approach. If the use of multimodal conversational interfaces expands in future, it will become indispensable to account for not only how people communicate with each other, but also how they communicate with computer artifacts. This thesis represents a contribution to this domain of interdisciplinary research.

Contents

1 Introduction **1**

1.1 Architecture of multimodal conversational interface 2
1.2 Human speech processing model 4
1.3 Interdisciplinary approach on designing conversational interfaces . . 4
1.4 Steps of our research method 5
1.4.1 Empirical study 6
1.4.2 Establishing a model 7
1.4.3 Evaluation 7
1.5 Contributions of the Thesis 7

2 Fundamental Work **11**

2.1 Speech act theories and plan-based systems 11
2.1.1 Speech act theories 11
2.1.2 Plan-based systems processing speech acts 12
2.2 Theories of discourse structure and discourse generation systems . . 13
2.2.1 Theories of discourse structure 13
2.2.2 Applying discourse theories to conversational systems 15
2.3 Theories of interaction structure and dialogue generation systems . . 16
2.3.1 Theories of interaction structure 16
2.3.2 Dialogue generation systems 16
2.4 Grounding and computational models of grounding 17
2.4.1 Clark's objection to previous discourse theories 17
2.4.2 Grounding 17
2.4.3 Computational model of grounding 18
2.4.4 Representing a state of grounding 20
2.5 Multimodal communication and multimodal interfaces 20
2.5.1 Communication through different communication modalities . 20
2.5.2 Multimodal interfaces 22
2.6 Motivation for empirical approach 22

3 Decision and Generation of Utterance Contents in Conversational Interfaces **25**
3.1 Study 1: Deciding Appropriate Query Content According to Topic Features . . . 25
3.1.1 Problem . . . 26
3.1.2 Describing Topic by Topic Features . . . 27
3.1.3 Utterance Content Unit and Query Structure . . . 29
3.1.4 Utterance Content Planner (UCP) . . . 31
3.1.5 Example . . . 35
3.1.6 Evaluation of Utterance Content Planner (UCP) . . . 36
3.2 Study 2: Factors for deciding utterance contents in instruction dialogues 37
3.2.1 Previous work . . . 38
3.2.2 Analysis of Instruction Dialogues . . . 39
3.3 Planning instruction dialogues . . . 45
3.3.1 Planning mechanism . . . 45
3.3.2 Heuristics for plan selection . . . 47
3.3.3 Example of instruction dialogue . . . 51
3.4 Summary . . . 53

4 Generating Multimodal Instruction Dialogues **55**
4.1 Overview . . . 56
4.2 Problems . . . 57
4.3 Previous work . . . 58
4.4 MID-3D System Architecture . . . 59
4.5 Selecting the Content of Instruction Dialogue . . . 61
4.5.1 Content Planner . . . 61
4.5.2 Considering the User's View in Content Selection . . . 61
4.6 Managing Interruptive Subdialogue . . . 62
4.6.1 Maintaining the Discourse Model . . . 62
4.6.2 Considering the User's View in Coping with Interruptive Subdialogues . . . 63
4.7 Example . . . 64
4.8 Summary and Discussion . . . 66

5 Dialogue Management Using Nonverbal Signals **67**
5.1 Problem . . . 68
5.2 Previous work . . . 70
5.2.1 Models of Grounding . . . 70
5.2.2 Nonverbal information as evidence of understanding . . . 71
5.2.3 Visual Information in Mediated Communication . . . 72

5.2.4 Nonverbal Behaviors of Animated Agents . . . 73
5.2.5 Our Approach . . . 74
5.3 Empirical Study . . . 74
5.3.1 Experiment . . . 75
5.3.2 Data Coding . . . 75
5.3.3 Analysis . . . 78
5.4 A Model of Face-to-Face Grounding . . . 86
5.5 Face-to-face Grounding with ECAs . . . 88
5.5.1 System Overview . . . 88
5.5.2 Nonverbal Inputs . . . 90
5.5.3 The Dialogue Manager . . . 90
5.5.4 Example . . . 94
5.6 Preliminary Evaluation . . . 95
5.6.1 Procedure . . . 96
5.6.2 Results . . . 96
5.6.3 Discussion . . . 97
5.7 Summary and Discussion . . . 98

6 Generating Gestures for Presentation Agents 101
6.1 Problem . . . 102
6.2 Background . . . 102
6.3 SPOC . . . 104
6.3.1 SPOC System Architecture . . . 105
6.4 Editing SPOC Contents . . . 106
6.5 CAST . . . 107
6.5.1 Background . . . 107
6.5.2 Linguistic Theories and Gesture Studies . . . 108
6.5.3 Empirical Study . . . 110
6.5.4 System Implementation . . . 112
6.6 Structure of a Knowledge Card . . . 115
6.7 Viewing SPOC Contents . . . 117
6.7.1 Automatic Camera Work Generation . . . 118
6.7.2 Playing Agent Animation . . . 119
6.8 Evaluation . . . 120
6.8.1 Experiment 1 . . . 120
6.8.2 Experiment 2 . . . 122
6.9 Related work . . . 124
6.9.1 Methods for multimedia contents generation . . . 124
6.9.2 Web-based Presentation Agent . . . 125
6.10 Summary and discussion . . . 125

7 Discussion and Future Direction **127**
7.1 Overall discussion 127
7.2 Evaluation scheme 128
7.3 Diversity and universality in communicative behaviors 128
7.4 Future directions 129
7.4.1 Producing contents for multimodal communication 129
7.4.2 Improving the reality of communication 130
7.4.3 Other communication artifacts 131

8 Conclusion **133**

A Instruction of experiment in Chapter 5 **135**

List of Figures

1.1 MCI architecture 2
1.2 Levelt's human speech processing model 3
1.3 Research method 5
1.4 Overview of contribution of this thesis 8

2.1 Example of rhetorical relation 14

3.1 The Input and output of UCP 28
3.2 Topic feature structure 29
3.3 Example of topic feature structure 29
3.4 Structure of query 31
3.5 Utterance Content Planner 32
3.6 Account of additional information 41
3.7 Percentage of turn release form 43
3.8 Turn strategy for novel/redundant information 44
3.9 The architecture of the instruction generation system 45
3.10 Example utterance content plan operator 46
3.11 Example dialogue strategy plan operators 47
3.12 A partial instruction dialogue plan for the Practice phase 51
3.13 Example dialogues between the system and a user 52

4.1 Right angle 57
4.2 Left angle 57
4.3 The system architecture 59
4.4 Examples of content plan operators 61
4.5 Example of the state of a dialogue 64
4.6 Example of a dialogue with MID-3D 65

5.1 Human face-to-face conversation 68
5.2 Snapshot of experiment session 76
5.3 Example of coding NV status 77
5.4 Mean number of the four types of UUs per dialogue 78

5.5 Mean number of NV status shifts occurring in each type of UU . . . 79
5.6 Example of non-verbal acts in Acknowledgement 81
5.7 Example of non-verbal acts for Info-req and Answer 82
5.8 Example of non-verbal acts for Assertion 83
5.9 Relationship between receiver's NV and giver's next verbal behavior 85
5.10 Example of conversation with MACK 88
5.11 MACK system architecture 89
5.12 Process of grounding judgment 92
5.13 Example of user (U) interacting with MACK (M) 94
5.14 MACK with user 95
5.15 Interaction log for user-MACK conversation 97

6.1 SPOC functions and components 104
6.2 SPOC system overview 105
6.3 SPOC program editing window 106
6.4 Knowledge Card Editor 107
6.5 Example analysis of syntactic dependency. Underlined phrases are accompanied by gestures, and strokes occur at double-underlined parts. Case markers are enclosed by square brackets [ ]. 109
6.6 CAST architecture 112
6.7 Overview of CAST and SPOC 113
6.8 Example of CAST output 114
6.9 Example of a Card in XML format 116
6.10 Snapshot of SPOC Viewer 117
6.11 Camera work 118
6.12 RISA snapshots. (a) RISA is doing a beat gesture, (b) RISA is looking away, (c) RISA is pointing at the visual material display, and (d) RISA is doing a "big" iconic gesture. 119
6.13 Synchronization between audio and animation 120
6.14 Example of Card recipes 121
6.15 Results of subjective evaluation 123

7.1 Embodied conversational agent embedded in a background 131

List of Tables

2.1 DU state transition diagram proposed by Traum(1994) 19
2.2 Factors for characterizing communication modalities 20
2.3 Costs in communication 21

3.1 Kernel Selection Rules 32
3.2 Weight and Threshold in Discriminant Functions 34
3.3 The Results of Applying Discriminant Functions to the Topic of "ANNOUNCE SERVICE" 36

5.1 NV statuses 77
5.2 Salient transitions 80
5.3 Nonverbal signals by MACK 91
5.4 Grounding Model for MACK 92
5.5 Preliminary Evaluation 96

6.1 Summary of results 111
6.2 Questions about general impression on SPOC 123

Chapter 1

Introduction

One of the most ambitions challenges in computer science is the creation of artifacts able to communicate with humans in a natural way. If people could interact with such artifacts in the same way as they communicate with their peers face-to-face, computer systems would become much easier to use.

In the hope of contributing to this goal, research into multimodal interfaces has proposed rich media environments, where users could interact with computer systems through text, speech, and graphics. More recently, virtual environments have often been used as graphical interfaces. Such user interfaces express their contents in multiple forms, and are expected to be easier to use than keyboard- or speech-based systems.

However, richer media does not always mean better media. The more complex the media become, the more difficult they are to integrate consistently and coherently in terms of human cognition and communication. Clark (2003) pointed out that one essential aspect of face-to-face communication is the fact that it is established by linking linguistic messages to the perceived world. Therefore, multimodal interfaces should be able to support the human cognitive process of connecting perceived information and linguistic propositions. For example, graphical information should match the content of dialogue and nonverbal behaviors through an animated character should be synchronized with spoken language.

Based on the aforementioned motivation, this thesis focuses on issues of communication capabilities in multimodal interfaces, and will propose designs for a Multimodal Conversational Interface (MCI) including multiple communication modalities in the form of speech, graphics and animations as well as conversational capability using natural language.

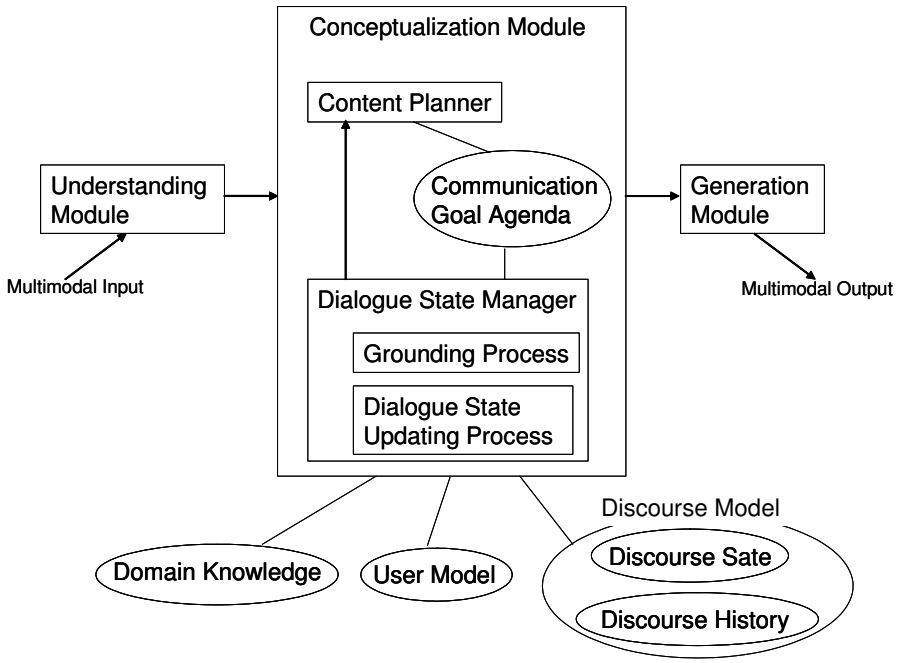


Figure 1.1: MCI architecture

1.1 Architecture of multimodal conversational interface

As the first step in designing an MCI, we start with defining the MCI architecture upon which the individual system proposed in the following sections is built. While a number of conversational systems have been developed so far, most of the systems employ similar architecture, consisting of main three functions: understanding the user's input, managing the conversation, and generating the system's output (Allen et al., 1996; Allen, Ferguson, and Stent, 2001). Extending this basic architecture, we propose our MCI architecture as shown in Figure1.1.

Once the user's input is received, it is sent to the *Understanding Module* to interpret the user's intention, such as requesting the system to provide information, providing information as an answer to the system's question, etc.

The next step is sending the system's interpretation to the *Conceptualization Module* (CM), which decides the system's response, namely what to say or do next. This decision is made by the *Content Planner*. The *Content Planner* refers to the *Communication Goal Agenda*: a list of goals which should be accomplished in a conversation. When all the goals in the agenda are accomplished, the interaction is successfully finished.

More importantly, in order to decide the content of the system response appropriately, the CM needs to know what is going on in the interaction and the *Dialogue*

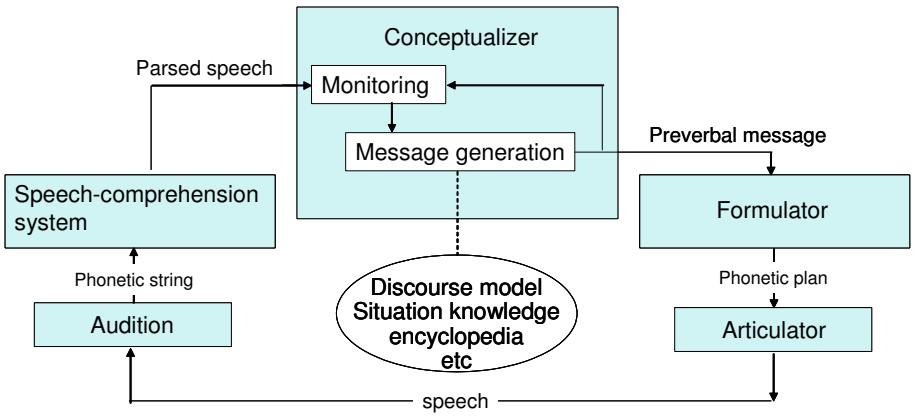


Figure 1.2: Levelt's human speech processing model

State Manager (DSM) is used for this purpose. It keeps tracks of the state of the conversation, and updates the state when necessary.

In the DSM, the *Grounding process* judges whether an understanding of what has been said has been successfully shared between the user and system. Then, the result of the grounding judgment is added to the *Discourse Model* through the *Dialogue State Updating Process*. This process updates the *Discourse State* representing the current state of conversation, and the Discourse History, which is a record of the past interaction.

The *Domain Knowledge* is a knowledge base of a specific task performed or discussed in the conversation. The *Domain Knowledge* may also be an expert system or AI-based reasoning system while the *User Model* stores information specific to the individual user. It may be a user's profile, knowledge level concerning the domain, or interests, etc. These two databases may both be used in the *Content Planner* and the *Dialogue State Manager*.

Finally, the output of the CM represents a concept as to what the system should say or do next. The concept is realized through the *Generation Module*, which constructs a natural language sentence and adds multimodal expressions, featuring graphics and animations, onto the linguistic expression.

As described above, an MCI is a relatively complex system consisting of many modules and sub-modules. There are an infinitive number of problems to be tackled, and building a complete system is far too ambitious a project to be undertaken in one thesis. Instead, since the *Conceptualization Module* (CM) plays a primary role in progressing the conversation, this thesis mainly addresses issues concerned with the CM.

1.2 Human speech processing model

In order to propose a model which is valid from a computational as well as a psychological point of view, it would be useful to compare a model of human speech processing with the MCI architecture.

Levelt (1989) explained human speaking ability by modeling a human speaker as an information processing mechanism as shown in Figure1.2. He claimed that talking as an intentional activity involves the conceiving of an intention, selecting the relevant information to be expressed, ordering this information for expression, keeping track of what was said before, and so on. He termed these mental activities *conceptualizing*, and the processing system the *Conceptualizer*.

Message generation in the Conceptualizer is a structured system consisting of condition/action pairs. For example, IF the intention is to commit oneself to the truth of p, THEN assert p. A set of these pairs works as *procedural* knowledge in generation process in the Conceptualizer.

The second type of knowledge used in speaking is *declarative* knowledge. In addition to the speaker's knowledge of the world (called *encyclopedic knowledge*), the speaker also has knowledge of the present discourse situation. Moreover, the speaker can access perceptual (e.g., visual and acoustic) information concerning the environment and objects within it. He called this type of knowledge *situational knowledge*. Finally, the speaker keeps track of what they themselves have said as well as what the conversational partners have said, and stores the course of interaction in the *discourse record*.

1.3 Interdisciplinary approach on designing conversational interfaces

Intriguingly, there is a beautiful symmetry in the relationship between the MCI architecture and the Levet's information processing model of human speech. Levet's *Conceptualizer* has very similar functions to those of *Conceptualization Module* in the MCI.

This similarity suggests that the MCI architecture can provide a psychologically as well as technologically valid system design, and that results of empirical studies conducted using psychological methods may contribute toward designing an MCI component. Upon the basis of the discussion above, we can define the following issues to be addressed in this thesis.

1. What are the determinant factors for selecting communicative behaviors within human communication, and how are these factors used to select message concepts in the Content Planner ?

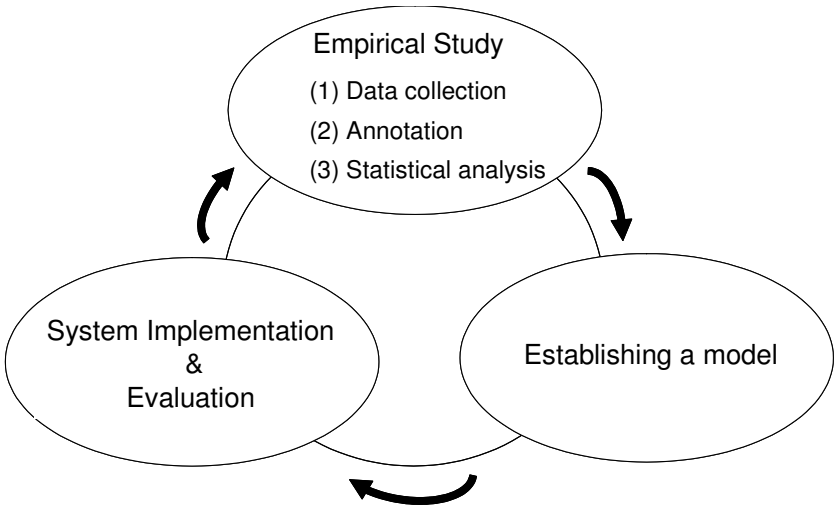


Figure 1.3: Research method

2. What types of information are important in representing the state of conversation in human communication, and how can these types of information be stored in the Discourse Model and used in the Dialogue State Updating mechanism in MCI ?

3. How is the common ground established in human communication, and how can this process be implemented in the MCI grounding mechanism ?

4. How do people communicate face-to-face using gestures and facial expressions, and how can an MCI mimic the communication using multimodal expressions ?

In the next section, we will describe the research method we used to solve these problems.

1.4 Steps of our research method

In order to design each component in the MCI architecture based on a psychologically valid model, we employ a spiral research approach shown in Figure1.3. First, we need to look at real human communication behaviors in establishing a model. Empirical studies are useful for identifying determinant factors for deciding what to say and/or do, controlling turn-taking, and changing the background situation, etc. Then, based on the results of the empirical study, factors predicting target behaviors are identified and a psychologically valid computational model is established. The next step is to implement the model as a component of MCI. When it is successfully implemented,

the system should be evaluated through a user study, for which the psychological method is useful. Studies through Chapters 3 to 6 are conducted by employing this method and the following section describes each step of this method in further detail.

1.4.1 Empirical study

(1) Data collection

The first step is to collect data concerning human communicative behaviors. In order to analyze this data statistically, it is necessary to elicit specific types of human behaviors and conversational phenomena by controlling the effects of the physical and social contexts where the conversation takes place.

Methods of experimental psychology are useful for this purpose. Equal numbers of subjects are assigned to each experimental condition: a situation of performing a conversation. In collecting task-oriented dialogues, the topic of conversation is also provided by the experimenter.

The same approach is also employed in the HCRC Map Task Corpus project (Anderson et al., 1991) followed by the Chiba Map Task Dialogue Corpus Project in Japanese (Horiuchi et al., 1999).

(2) Annotation of communicative behaviors and phenomena

The next step is to specify the type of verbal and nonverbal behaviors and phenomena to be investigated and annotate them in the data. Verbal behaviors would be categorized according to the type of syntactic structure, speech act, as well as the rhetorical relation constructing the discourse. A sequence of interaction, such as turn taking, is also frequently annotated in conversational data. Nonverbal behaviors are categorized according to the physical movement of a specific part of the body (Ekman and Friesen, 1969; Bull, 1987; McNeill, 1992). Facial expressions, hand gestures, and postures are known to be meaningful body movements correlated with verbal behaviors.

(3) Statistical analysis

The collected data are then analyzed statistically. In many cases, the frequencies for specific types of behaviors are counted and the average frequencies for each experimental condition are calculated as basic statistics characterizing the data. Analyzing the correlation between different types of behaviors, such as verbal and nonverbal behaviors, is also often useful for finding relationships between types of behaviors independently annotated.

1.4.2 Establishing a model

The next step is to establish a model based on the results of the empirical study and use this model to predict specific behaviors. For example, if a specific type of gesture frequently co-occurs with a specific type of linguistic expression, occurrence of that type of gesture can be predicted by the type of linguistic expression used.

Note that the model should be formal and executable in nature so that it can be implemented into a computer system, meaning mathematical or statistical models based on the empirical results are preferable. A decision algorithm described in form of an IF-THEN rule is also useful as a computational model. Within a plan-based system, describing preferences or constraints on plan selection based on the empirical results would contribute to the establishment of a psychologically valid planning mechanism.

1.4.3 Evaluation

When the model is implemented into a system and works effectively within an entire conversational system, the final step of the research is to examine whether the implemented mechanism actually contributes to improving the interaction between users and systems. Experimental methods similar to those employed in collecting the data can be used for evaluating the system.

1.5 Contributions of the Thesis

The main contribution of this thesis is to establish interdisciplinary research on multimodal conversational interface. The notable characteristic of our approach is to seamlessly connect an empirical study with a system design. Thus, the study consists of several steps: empirical study to learn from real human behaviors, establishing a computational model based on the results of the empirical study, system construction and finally evaluation.

Although this sequence may seem obvious, few studies have accomplished all these steps in a consistent manner. Most empirical studies are interested only in analyzing human behaviors and provide explanatory models which rarely make a direct contribution to MCI designs. On the other hand, most multimodal systems are designed without an empirical basis of real human communication behaviors.

With the goal of designing more natural human interfaces, this thesis will show how empirical studies conducted using psychological research methods can directly contribute to MCI design implemented by media technologies. Moreover, evaluation experiments will show how these models improve human-computer interaction.

Figure 1.4 depicts the outline of the thesis, and shows that individual study contributes to a module in MCI architecture.

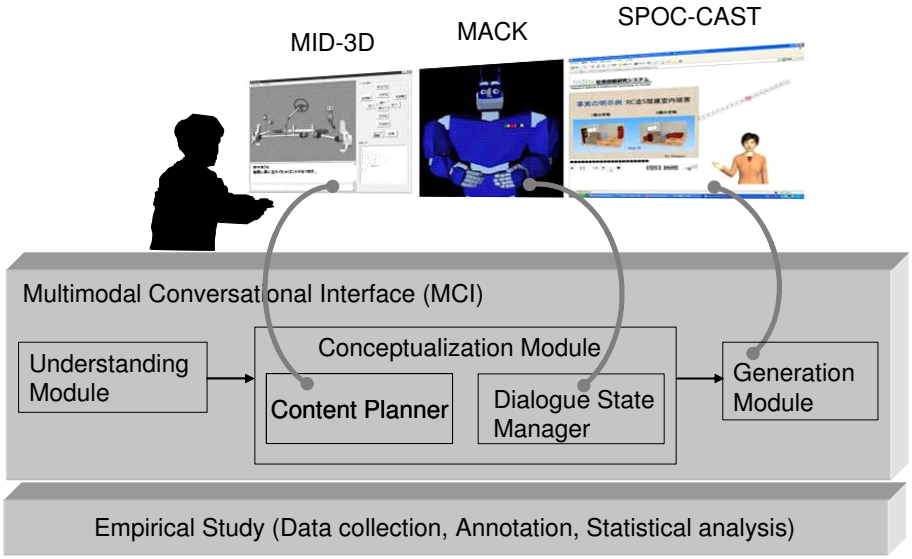


Figure 1.4: Overview of contribution of this thesis

Chapters 3 and 4 address issues on the Content Planner. In Chapter 3, we start with analyzing service reception dialogues and then establish a model to determine appropriate query content according to the characteristics of the topic. For example, when asking the name of the customer, the receptionist simply asks the core question, "Your name please". On the contrary, when asking for contact information, s/he mentions the reason for asking the question in addition to the core question, for example: "Next, we need your contact telephone number. We may need to contact you. What number may I call ?" Based on the data analyses, we will establish an algorithm deciding when and what type of core question and which additional information should be selected according to the topic characteristics represented as a quintuple of feature values.

In the second part of chapter 3, we analyze instruction dialogues for installing an answering machine. We define three types of dialogue state, namely Practice, Review, and Re-explanation, according to the number of task trials and whether the learner failed or succeeded in accomplishing the task in the last trial. Then, we investigate the way in which the instructor's utterance contents differ depending on the type of dialogue state used. In system implementation, we employ a planning mechanism to determine the explanatory utterance contents for instructing operations, and propose heuristics for selecting plan operators based on the results of the dialogue data analysis.

Chapter 4 modifies the plan-based Content Planner described in Chapter 3, to generate multimodal instruction dialogues. We will propose a mechanism for alter-

ing the instruction dialogue to match the user's view in a virtual environment, and implement a dialogue generation mechanism in the MID-3D system, which interactively instructs the user on dismantling certain car parts. Moreover, this chapter employs a stack-based discourse model as proposed by (Grosz and Sidner, 1986), so that it can cope with interruptive subdialogues initiated by the user.

Chapter 5 addresses issues on the Dialogue State Manager in the MCI. As one of the most important issues in terms of multimodal dialogue management, this chapter focuses on face-to-face grounding, more specifically on nonverbal signals used for such face-to-face grounding. We will analyze how verbal and nonverbal behaviors (e.g., eye-gaze and head nod) interact with each other in grounding, and identify positive and negative nonverbal evidence of grounding. We will then implement the nonverbal grounding model into a conversational kiosk agent, MACK. Although there is a small frog-like character in MID-3D in Chapter 4, in this chapter, we improve the character agent animation allowing the expression of facial expressions as well as the display of gestures.

Chapter 6 will propose a mechanism for the Generation Module, which generates the final output from the MCI. Although a multimodal generation mechanism should be able to generate multiple types of media, such as speech, text, graphics, movies, and animation, in a synchronized manner, this chapter focuses on the synchronized generation of agent animation and speech. First, we will analyze human gestures observed in presentations, and investigate how gesture occurrence and linguistic information interact with each other. Then, based on the empirical study, we will implement an agent behavior generation system, CAST, and integrate it into an automatic presentation generation system, SPOC. Since SPOC is a web-application, SPOC-CAST suggests the possibility of building a web-based MCI for practical usage, and providing a service to be used by anyone with network access.

As described above, this thesis addresses a wide range of issues concerned with the building and design of MCI, and mainly contributes to research into multimodal content generation and dialogue state management. Finally, note that our research method, which acts as a bridge between empirical studies and system building, is a highly original and strong point of this thesis.

Chapter 2

Fundamental Work

This chapter describes an overview of the previous research based on which studies of the following chapters have been built. Section 2.1 presents an introduction to speech act theory, and how this theory has contributed to the primary stage of research on collaborative dialogue systems. Section 2.2 describes theories of discourse, and then shows how these theories have been used in representing discourse models and generating discourse in conversational systems. Section 2.3 presents a hierarchical model of conversation proposed by Discourse Analysis, and findings reported by Conversational Analysis. Section 2.4 addresses issues on grounding. First, we describe the theoretical background, and then show how computational models of grounding are implemented as a dialogue management mechanism. Section 2.5 describes studies of multimodal communication, and reviews recent studies of multimodal systems which have been proposed at the same time of this thesis work. Finally, on the basis of this review, we discuss why empirical-based approach is necessary.

2.1 Speech act theories and plan-based systems

2.1.1 Speech act theories

In conversation, the act of uttering words is used for communicating with the conversational partners. Austin (1962) pointed out that utterances are not used just to describe states of affairs, but rather actively to *do* things. He observed that there are three types of acts performed whenever something is said:

Locutionary acts: act of uttering a sequence of words

Illocutionary acts: act that the speaker performs in saying the words

Perlocutionary acts: act that actually occurs as a result of the utterance

All utterances, in addition to meaning whatever they mean, perform specific actions (or "do things") through having specific forces. For example, when a speaker says, "Shut the door" in appropriate circumstances, it has the *illocutionary force* of ordering the interlocutor to shut the door.

Searle (1975) pointed out that the relation between speech acts and the devices used to indicate them is complicated. For example, there are various ways of requesting an addressee to shut the door:

a. I want you to close the door.

b. Can you close the door?

c. Would you mind closing the door?

d. It might help you close the door.

e. It's cold here.

He claimed that indirect illocutionary force can be calculated from the literal meaning of an utterance. In case **e**, "It's cold here" can function as request to shut the door because the speaker's intention can be derived from an assertion that the temperature is low.

2.1.2 Plan-based systems processing speech acts

A plan-based model of speech acts was suggested by (Bruce, 1975) and developed in a series of papers, where a speech act is used as a unit of verbal behavior for exchanging intention between a user and a system, starting with one by Cohen and Perrault (1979). This paper lays out the general principles of the approach and shows how speech acts can be planned in order to achieve goals using standard planning techniques. Perrault and Allen (1980) and Allen and Perrault (1980) proposed a computational model of indirect speech acts using plan recognition, and described how indirect speech acts can be computed from a literal meaning based on the work by (Searle, 1975).

Littman and Allen (1987) and Carberry (1990) extends Allen and Perrault's work to include dialogues rather than just single utterances, and have a hierarchy of plans rather than just a single plan. They describe two different types of plans: domain plans and discourse plans. Domain plans are those used to perform a cooperative task, while discourse plans, such as *clarification* and *correction*, are used to plan a course of the interaction, and are task-independent.

2.2 Theories of discourse structure and discourse generation systems

While speech act is a unit of verbal action, a sequence of utterances or sentences connected coherently constructs a discourse (e.g., explanation text, spoken instruction, and task-oriented conversation). Therefore, in addition to a unit of verbal communication, theories of discourse are necessary to describe a structure of a set of utterances or sentences, and mechanisms of processing a discourse also need to be developed based on theories of discourse.

2.2.1 Theories of discourse structure

Grosz and Sidner's framework

Grosz and Sidner (1986) proposed three distinct but interrelating components of discourse structure.

1. Linguistic structure
2. Intentional structure
3. Attentional state

The first component of discourse structure is the structure of the actual sequence of utterances that comprise a discourse. The utterances are naturally aggregated into *discourse segments*. The utterances in a segment, like words in a phrase, serve particular roles with respect to that segment. It is frequently observed that certain kinds of words and phrases indicate discourse segment boundaries. These kinds of expressions are called *cue phrases* because each one of these devices cues the interlocutor to some change in the discourse structure. In addition to the boundaries between discourse segments, the linguistic structure consists of embedding relationships between the segments.

The second component is the intentional structure. A discourse has an overall purpose, which is called *discourse purpose (DP)*, and each of the discourse segments has a single intention, which is called *discourse segment purpose (DSP)*. If an intention is a DP, then its satisfaction is a main purpose of the discourse, whereas if it is a DSP, then its satisfaction contributes to the satisfaction of the DP. What is essential for discourse structure is that such intentions bear certain kinds of structural relationships to one another.

The third component of discourse structure is the attentional state. This is an abstraction of the participants' focus of attention as their discourse unfolds. The attentional state is modeled by a set of *focus spaces*. A focus space is associated with each discourse segment, which includes a DSP. A collection of focus spaces

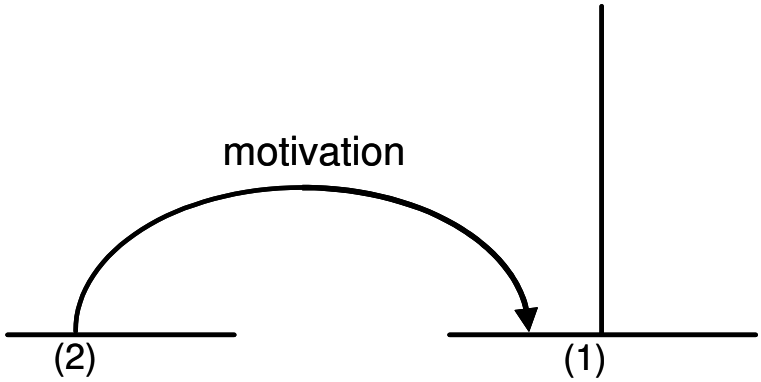


Figure 2.1: Example of rhetorical relation

is called a focusing structure or focus stack, which has pop and push operations. The segment on the top of the stack is the current most salient discourse segment. A push occurs when the DSP for a new segment contributes to the DSP for the immediately preceding segment. When the DSP contributes to some intention in higher level segments, several focus stack spaces are popped from the stack before the new one is inserted.

Rhetorical structure theory

Aiming at describing the connection between rhetorical relations and speaker intentions, two theories of discourse structure have been proposed: Hobbs' (Hobbs, 1979) theory of coherence relations and Mann and Thompson's (Mann and Thompson, 1987) *Rhetorical Structure Theory* (RST). RST can be adapted to a computational model in a fairly natural way, and in fact there is an implemented prototype of the theory (Hovy, 1975).

The definition of each *rhetorical relation* in RST indicates constraints on the two entities being related, constraints on their combination, and a specification of the effect that the speaker is attempting to achieve on the hearer's beliefs or inclinations. Thus, RST provides an explicit connection between the speaker's intention and the rhetorical means used to achieve it.

An RST relation has two parts: a *nucleus* and *satellites*. The nucleus of the relation is essential to express the speaker's purpose. The satellites represent supporting information. For example, (1) and (2) are related by MOTIVATION relation.

(1) Come to the party for the new president. (2) There will be lots of good food.

As shown in Figure 2.1, in this example, (1) is the nucleus indicating that the

speaker's intention is to make the hearer go to the party, and (2) is the satellite that represents the support for (1).

2.2.2 Applying discourse theories to conversational systems

Representing a discourse model

Collagen (Rich and Sidner, 1998) provides dialogue management mechanism, which keeps track of the state of conversation, based on (Grosz and Sidner, 1986; Lochbaum, 1998; Sidner, 1994). Collagen updates the focus stack and plan tree using a combination of the discourse interpretation algorithm of (Lochbaum, 1998) and plan recognition algorithms of (Lesh, Rich, and Sidner, 1999). It takes as input user and system utterances and interface actions, and accesses a library of recipes describing actions in the domain. After updating the discourse state, Collagen makes three resources available to the interface agent: focus of attention (using the focus stack), segmented interaction history (of completed segments) and an agenda of next possible actions created from the focus stack and recipe tree.

Generating interactive explanation

In generating a discourse, a system needs to select information relevant to achieving a specific discourse purpose and organize the information as a coherent text that achieves the purpose.

In her text generation system, TEXT, McKeown (1985) devised several script-like (Schank and Abelson, 1977) structures, called schemata, which represent combinations of rhetorical predicates. By associating each rhetorical predicate with an access function for an underlying knowledge base, these schemata can be used in a text generation process.

Moore (1995) pointed out that McKeown's schemata lack an explicit representation of the intentional structure of the text being produced. Moreover, schemata are too rigid to handle certain of the opportunistic phenomena observed in naturally occurring explanations. In order to overcome these problems, Moore (1995) proposed to employ plan operators to encode knowledge about how communicative intentions may be achieved via a set of rhetorical relations. For this purpose, they defined a plan language that links intentions to the rhetorical means for achieving them.

2.3 Theories of interaction structure and dialogue generation systems

2.3.1 Theories of interaction structure

In Discourse Analysis, Sinclair and Coulthard (1990) and their followers (Stenström, 1994) proposed a model of tutorial discourse. The top level is *transaction*, which is made up of *exchanges*. Each *exchange* consists of *turns*. Then, each *turn* consists of *moves*, while each *move* contains more than one *act*.

By contrast, Conversation Analysis (Levinson, 1983), pioneered by a group of sociologists (often known as ethnomethodologists), is against the structuralism in studying conversation. It shows how the functions that utterances perform are in large part due to the place they occupy within specific conversational (or interactional) sequences, or contexts.

As one of the most important findings by Conversational Analysis is *turn-taking*: one participant, *A* talks, stops; another, *B* starts, talks, stops; and so we obtain an A-B-A-B distribution of talk across two participants (Sacks, Schegloff, and Jefferson, 1974). There is no predetermined structure for how long a particular turn will last, but there are locally organized principles for shifting turns from one participant to another.

Another local management organization in conversation is *adjacency pair*: paired utterances such as question-answer, greeting-greeting, offer-acceptance, etc. Once a speaker has produced a first part of an adjacency pair, s/he must stop speaking, and next speaker must produce a second part of the pair. Therefore, adjacency pairs are deeply inter-related with the turn-taking systems.

In addition to the local organization, they also proposed overall organizations of conversation: *opening* and *closing* of conversation. For example, greetings are used in opening of conversation, and farewells are contained in closing.

2.3.2 Dialogue generation systems

Cawsey (1990) employed these theories of interaction structure to design an instruction dialogue system. She developed the EDGE system, which gives explanations about different types of electric circuits. In order to generate an interactive explanation, she employed AI planning technique and defined plan operators based on the hierarchical model of conversation in Discourse Analysis: *transaction*, *exchange*, *turn*, *move*, and *act*, and ideas of conversational organization in Conversational Analysis: *turn taking*, *adjacency pair*, and *opening-closing*. Her system also has a model of discourse structure based on Grosz and Sidner (1986), and plan schemas, which are used to construct explanations.

2.4 Grounding and computational models of grounding

We described research on speech acts, discourse structure, and interaction structure, all of which contribute to representing discourse and intentions behind them. Finally, we need to describe how conversational participants establish a conversation and accomplish shared knowledge through the conversation.

2.4.1 Clark's objection to previous discourse theories

Classical theories of discourse presuppose the following three points concerning common ground in discourse.

Common ground: the participants in a discourse presuppose a certain common ground.

Accumulation: in the course of discourse, the conversational participants add shared knowledge to their common ground.

Unilateral action: common ground is added by a speaker uttering the right sentence at the right time.

Clark objected to the third assumption. He claimed that this assumption is not sufficient to handle conversation because these theories are only concerned with a speaker's intention, and assume that what the speaker said is added to the discourse model without any error. The previous theories of discourse were not concerned with dynamics in conversation, and operated on the strong assumption that the hearer understands rationally, and that a speaker's utterance is perfectly understood by the hearer if it is rationally appropriate.

As an extension of this discussion, Walker (1992) proposed IRU (information redundant utterance), which is an utterance that does not add new propositions in the discourse. She claimed that repeating what the speaker has already said is informationally redundant, but this kind of utterance provides evidence that the mutual understanding is actually achieved.

2.4.2 Grounding

Grounding is a process to make what has been said a part of common ground. Clark and Schaefer (1989) proposed a model for representing grounding using a concept of "contributions". In their model, a contribution is composed of two main phases.

Presentation Phase: A presents utterance u for B to consider. He does so on the assumption that, if B gives evidence e or stronger, he can believe that B understands what A means by u.

Acceptance Phase: *B* accepts utterance *u* by giving evidence *e'*, that he believes he understands what *A* means by *u*. He does so on the assumption that, once *A* registers evidence *e'*, he will also believe that *B* understands. Through these two phases, people in conversation contribute to discourse to reach the grounding criterion(Clark and Schaefer, 1989).

In addition to these basic processes for grounding, they proposed a notion of "grounding criterion". The basic idea is that the contributor and the partners mutually believe that the partners have understood what the contributor meant to a criterion sufficient for current purposes. In achieving common ground, it is often not necessary to assure perfect understanding of each utterance but only understanding "to a criterion sufficient for current purposes" (Clark and Schaefer, 1989). Therefore, we can have some sort of common ground without full mutual belief, and also the grounding criterion may change as conversation purposes change.

As grounding criterion is different depending on the purpose of conversation, it is possible to define different levels of conversation according to how perfectly a listener hears or understands a speaker's presentation. Based on the previous studies (Allwood, Nivre, and Ahlisen, 1992; Clark, 1994), Traum and Dillenbourg (1998) proposed four different levels of conversation in terms of maintaining common ground in multimodal environment, which is shown below.

Level 1. access : do the collaborators have access to each others communicative actions

Level 2. perception : do the collaborators perceive the communicative actions that are performed

Level 3. understanding : do the collaborators understand what is meant by the communicative actions

Level 4. agreement : do the collaborators reach agreement about the communicated facts of plans

2.4.3 Computational model of grounding

Traum (1994) proposed a computational model of grounding. He proposed *grounding act*, which could be realized with a single utterance or a speech act, as a unit of contribution to grounding. Rather than the two phases of presentation and acceptance, the basic building blocks are a set of grounding acts, each of which is identified with a particular utterance unit, and performs a specific function towards the achievement of common ground.

In this model, the units of grounded content are called Discourse Units (DUs), rather than Contributions. Individual grounding acts could add or change content

Table 2.1: DU state transition diagram proposed by Traum(1994)

Next Act	In State						
	S	1	2	3	4	F	D
Initiate^{I}	1						
Continue^{I}		1			4		
Continue^{R}			2	3			
Repair^{I}		1	1	1	4	1	
Repair^{R}		3	2	3	3	3	
ReqRepair^{I}			4	4	4	4	
ReqRepair^{R}		2	2	2	2	2	
Ack^{I}				F	1*	F	
Ack^{R}		F	F*			F	
ReqAck^{I}		1				1	
ReqAck^{R}				3		3	
Cancel^{I}		D	D	D	D	D	
Cancel^{R}			1			D	

of the unit. Based on this claim, he proposed a DU state transition diagram, which defines possible sequence of grounding acts to achieve common ground. In Table 2.1, S stands for start initial state and F for final state. D stands for dead state, where the conversational material can no longer be grounded. The network is traversed by observing grounding acts as shown in each row in the table. For example, in the start state, an Initiator (I) initiates a new DU, and the state moves to *State 1*. If a Responder (R) requests (I) to repair the statement at *State 1*, (I) needs to repair, or requests (R) to repair. This is *State 2*. If (I) repairs at *State 2*, the next state is back to *State 1*. Then if (R) returns an acknowledgement at *State 1*, the DU is grounded (the final state).

There are some other studies for computational models of grounding. Heeman and Hirst (1995) presented a computational model for grounding a referring expression. They employed a planning paradigm in modeling how conversational participants collaborate in making a referring action successful as well as clarifying a referring expression.

Paek and Horvitz (1999) claim that the majority of previous dialogue systems focus only on the intention level, but it is necessary for a dialogue system to handle other levels of conversation. They provided infrastructure that recognizes conversation failures happening at any levels of conversation, and proposed representations and control strategies for grounding using Bayesian networks and decision theory. Based on the four levels of conversation proposed originally by (Clark and Schaefer, 1989), they employed these representations and inference strategies at four levels; Channel level, Signal level, Intention level, and Conversation level.

Table 2.2: Factors for characterizing communication modalities

Modality	Factors
Face-to-face	Copresence, visibility, audibility, cotemporality, simultaneity, sequentiality
Telephone	audibility, cotemporality, simultaneity, sequentiality
Video teleconference	visibility, audibility, cotemporality, simultaneity, sequentiality
Terminal teleconference	cotemporality, sequentiality, viewability
Answering machine	Audibility, reviewability
Electric mail	Reviewability, revisability
Letters	Reviewability, revisability

2.4.4 Representing a state of grounding

TrindiKit (Larsson et al., 1999) is a toolkit which provides support for developing dialogue systems, focusing on the central dialogue management components. A prominent feature of TrindiKit is the *information state*, which serves as a central "blackboard" that is subject to various kinds of update mechanisms. The structure of the information state can be customized by the system developer. For example, Matheson, Poesio, and Traum (2000) implemented their dialogue processing engine using TrindiKit to support a grounding process in the dialogue system.

2.5 Multimodal communication and multimodal interfaces

2.5.1 Communication through different communication modalities

Human communication behaviors change dramatically according to the communication medium. As shown in Table 2.2, Clark and Brennan (1991) proposed seven ways in which a medium may affect the communication between two people. They also proposed various kinds of costs that change depending on the characteristics of the medium (Table 2.3).

They mentioned that, in face-to-face conversation, it is easy to nod at interlocutors, and to gaze at interlocutors to show them that they are being attended to, or to monitor their facial expressions. On the contrary, in media without co-presence, gestures cost expensive bandwidth, or are severely limited.

In a study reported by (Cohen, 1984) in which tutors instructed students on assembling a pump, they compared communication by telephone with one by keyboard. In a telephone conversation, producing an utterance and changing speakers does not cost much. On the other hand, in keyboard conversation, the cost for

Table 2.3: Costs in communication

- Formulation costs
- Production costs
- Reception costs
- Understanding costs
- Start-up costs
- Delay costs
- Asynchrony costs
- Speaker change costs
- Display costs
- Fault costs
- Repair costs

changing a speaker and repair cost are high. Therefore, subjects formulate utterances more carefully in keyboard conversation than in telephone conversation.

Brennan (2000) provides experimental evidence that reveals how grounding is accomplished in conversational tasks. She used a computer-based location task, where one party (the director) must describe where on a map the other (the matcher) is to point his cursor. This experiment is broken up into two conditions where the director can see where the matcher is vs. where the director cannot. In the second condition, the director must rely on verbal descriptions from the matcher. This experimental manipulation changes the strength and type of evidence available for accepting presentations. The results of the experiment revealed that the grounding process was shorter when more direct evidence (i.e., visual evidence indicating the place of the matcher's cursor) was available.

Dillenbourg, Traum, and Schneider (1996) analyzed grounding across different modes of interaction. They used a virtual environment that the subjects modified by giving on-line commands, such as redirecting the location of the character of the user. In their experiment, the subjects used three modes of communication: dialogue, action command in the virtual environment, and whiteboard drawing. In dialogue, the subjects talked to each other via two commands, "say..." to communicate with anybody in the same room, and "page <Player>..." to communicate with this player wherever he is. Using action commands, they changed the virtual environment, such as the location of the user or other objects. The third mode of communication, whiteboard drawing, was visible in the form of a non-scrollable window that remained in the subjects screen until it was deleted. By looking at cross-modal grounding, they found that grounding is often performed across different modes. For example, information presented in dialogue is grounded by an action in the virtual environment. Also, actions in the virtual environment are grounded

in the dialogue.

2.5.2 Multimodal interfaces

Attempting to mimic human multimodal communication, studies on multimodal interfaces started with multimodal presentation systems which automate the authoring and organizing process of different kinds of media, such as text, graphics, animation, and sound for the presentation information, to provide information. As multimodal presentations can be represented by similar principles to those for text organization, discourse generation techniques have applied to organizing a multimodal presentation.

COMET (Feiner and McKeown, 1993) and SAGE (Kerpedjiev et al., 1997) employed a notion of schemata based on a text generation mechanism proposed by (McKeown, 1985). As discussed in reviewing text generation research (Section 2.2.2), in order to overcome the problems of schema-based approach, plan-based approaches have become more popular (André, Rist, and Muller, 1999; Dahal et al., 1996; André and Rist, 1993). The idea behind these systems is to generalize communicative acts to multimodal acts and to formalize them as plan operators executed in a planning system (André, 2000).

More recently, thanks to a great advance of computer graphics, multimodal interfaces have been extended to learning environments where animated characters cohabit the environments with students to create rich, face-to-face learning interactions. Rickel and Johnson (1999) built a pedagogical agent embodied in a 3D virtual environment and demonstrates sequential operations of complex machinery and answers some follow-up questions from the student. Lester et al. (2000) developed a lifelike pedagogical agent, Cosmo, which can generate deictic behaviors and emotive-kinesthetic behaviors including facial expressions and gestures with arms and hands. This system provides advice to students about Internet packet routing.

Research on Embodied Conversational Agents focuses on communication capability of animated agents, and has implemented agents that can generate non-verbal behaviors such as head nod, gaze towards user and away, and gestures (Cassell et al., 2001). The goal of this approach is to improve naturalness of human-computer interaction by implementing face-to-face conversational protocols into animated agents.

2.6 Motivation for empirical approach

This chapter reviewed important topics and remarkable studies that provide the foundations of this thesis. A more detailed review of each topic will be presented in the individual chapters.

As described in the previous sections, linguistics and psychology present theories

for describing the characteristics of human communication while computational linguistics provides the models and mechanisms to handle linguistic communications with computer programs. Although implementation of an MCI would be possible by simply combining these two disciplines, we claim that better designs of MCI must take shape through consistent integration of a computational model and empirical support for the model derived from the analysis of real human communication behaviors. Note that although the small number of descriptive examples cannot itself provide sufficient empirical support. The sufficient empirical support should be able to be derived from statistical analysis using an effective amount of real human communication data.

This thesis therefore aims to render human-computer interaction through the MCI more natural by applying knowledge concerning human communicative behaviors to future MCI design. The following chapters will show how this goal is accomplished in designing and building each component of MCI architecture.

Chapter 3

Decision and Generation of Utterance Contents in Conversational Interfaces

This section addresses issues for selecting utterance contents in conversational interfaces. First, we conduct empirical studies in order to reveal the determinant factors for utterance content selection. In Section 3.1, we analyze questions in service reception dialogues. Section 3.2.2 analyzes explanatory discourse in instruction dialogues for installing an answering machine. Then, we propose a plan-based mechanism that determines appropriate explanatory utterance contents using heuristics or statistical models found in the empirical studies.

3.1 Study 1: Deciding Appropriate Query Content According to Topic Features

Dialogue Systems must generate appropriate queries because, in most service reception dialogues, it is necessary to extract information from the user as well as to give information to the user. While there are several factors that influence the appropriateness of queries in dialogues, in particular the characteristic of *topic* (what the speaker plans to ask about) is an important factor in determining the content of the query. This section focuses on the dependency of query content on topic feature. By observing real conversation data, five topic features are extracted. For each topic, the five topic features take specific values, therefore a topic can be represented as a quintuple of the topic feature values. Based on this framework, we propose the *Utterance Content Planner (UCP)* which selects the most appropriate query content according to the quintuple of topic feature values. UCP takes the topic feature val-

ues as input, and outputs a list of *Utterance Content units (UC units)* that represent the query contents. We evaluate the UCP with actual data to examine whether it can predict real conversational expressions quite well.

3.1.1 Problem

If a system interacts with humans by using natural language, it is crucial to query the users to acquire useful information as well as answer users' questions. Most studies in natural language processing, however, consider only how to appropriately and informatively answer the user's questions (Allen 1987; Webber 1987). We, on the other hand, consider the appropriateness of query usage since it is essential for the system to extract information from the user, especially in mixed initiative dialogues like consultations. In such a conversation, in order to recognize the users' needs and interests, the system must be able to extract information from them by using the most appropriate queries. This chapter considers how to query the users and proposes an algorithm that generates appropriate conversational queries in the domain of telephone operator and the customer conversation.

There are several aspects in generating appropriate discourses. As for the appropriateness of discourse structure, rhetorical information and attentional information are important because they contribute to discourse coherence. The RST schema (Mann and Thompson, 1987) and discourse focusing (Grosz and Sidner, 1986) can help to effectively generate coherent discourses (Moore and Paris, 1989; McKeown, 1985).

As for the appropriateness of discourse contents, it is most important how cooperative and informative the utterance to the user is. Research on this aspect mainly focused on responses to the users' questions. The system answers appropriately to the users' questions through the comprehension of the users' goals and beliefs by means of scripts (Lehert 1977) or goal (plan) interface techniques (Cohen and Perrault, 1979; Allen and Perrault, 1980).

There are, however, few studies about what to say in generating queries except for Bunt's work (1999) which generates appropriate queries based on appropriate conditions of the speakers' intentions and beliefs. In this section, we focus on generating cooperative and informative queries in conversational situation. By observing actual conversations and analyzing them statistically, we claim that appropriate query contents depend not only on beliefs and intentions or some linguistic factors, but also on the topic (topic characteristics). Topic is defined in our research as the item about which the speaker plans to ask: such as asking for a name, or asking for a contact number. A typical telephone conversation is given below.

Asking the customer's name:
Op: "Your name please."

Asking the customer's contact number:
Op: "Next, we need your contact telephone number.
We may need to contact you.
Either the number of your office or a relative is OK.
Would you tell me the appropriate number?"

The telephone operator (Op) is asking the caller for information regarding a change in residence. When asking for the customer's name, the operator usually uses a simple direct query, "Your name please." When asking a customer's contact number, the operator usually includes additional information, like "Next, we need your contact telephone number ... your office or a relative is OK." This observation indicates that the operator decides the utterance content depending on the characteristics of the topic. For instance, asking the customer's contact number is uncommon for the hearer and the reason for the request is hard to understand. In such cases, it seems necessary, to the operator, to supply additional information. Therefore, topic characteristics reflect how we anticipate the hearer will interact with the topic. We think that topic characteristics are one of the most important factors in deciding query content though there are other determinants of utterance content (Bunt, 1989).

Based on the idea mentioned above, we propose the Utterance Content Planner (UCP), and an algorithm which decides the query content according to the topic characteristics. As shown in Figure 3.1, UCP takes as input the quintuple of *topic features* that characterize the topic and influence query content, and outputs, a list of Utterance Content Units (UC units) as the query content. Section 3.1.2 defines the topic features and shows how topics are represented by feature values. Section 3.1.3, defines the Utterance Content Unit (UC unit). We also discuss the structure of queries. In section 3.1.4, we propose the Utterance Content Planner (UCP). An example is shown in section 3.1.5, and the UCP is evaluated in section 3.1.6.

3.1.2 Describing Topic by Topic Features

In order to formally represent the characteristics of topics, we defined five topic features which influence query content. They were derived from empirical studies (Ishikawa and Kato, 1991). These topic features are used in UCP to decide the appropriate utterance content. Each of these topic features has a value, and a topic is represented by a quintuple of feature values. This parameterization of topic make it possible to provide formal rules for deciding appropriate utterance content that are independent of the topic domain. The five topic features are defined below.

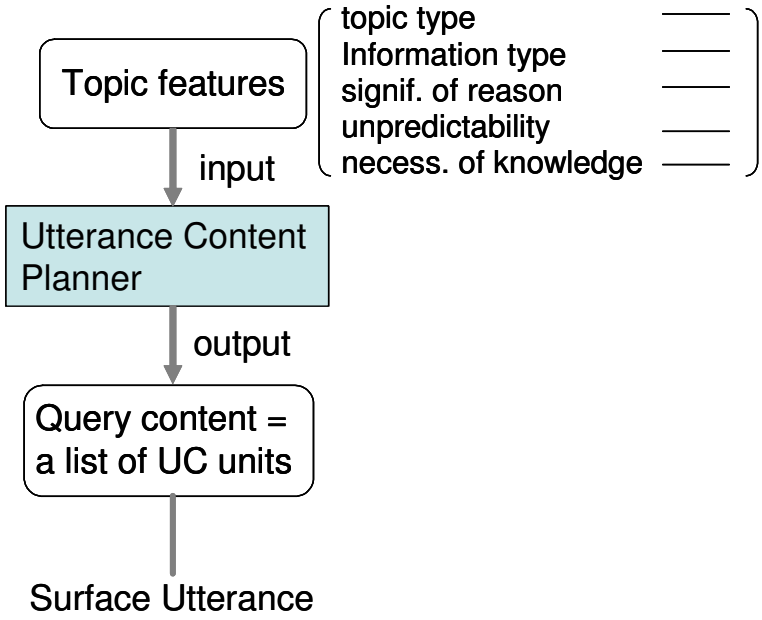


Figure 3.1: The Input and output of UCP

1. **Topic Type:** the value of this feature is either *fact* or *will*

   Fact : Asking for factual information such as name or address.

   Will: Asking about the hearer's will; which is preferred, or what action is desired.

2. **Information Type:** This feature has three possible values depending on what type of information is to be given.

   Yes/No: Answer yes or no.

   Selecting option: Answer by selecting one of the options.

   Descriptive explanation: The hearer should answer by describing or explaining in his/her own words.

3. **Significance of Reason:** This reflects how significant the reason for the query is, and how difficult the hearer finds it to understand the necessity of the topic. This feature takes a value between one to five.

4. **Unpredictability:** This addresses with how unexpected and surprising the topic is for the hearer. This feature takes a value between one to five.

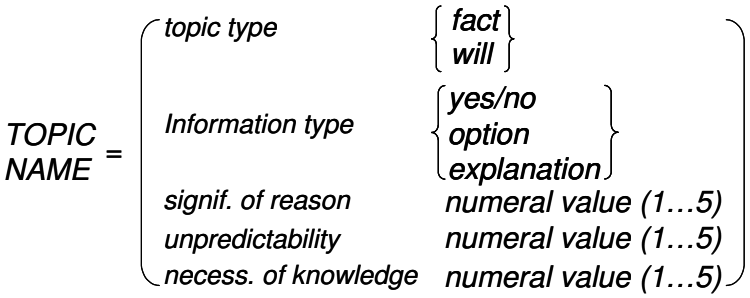


Figure 3.2: Topic feature structure

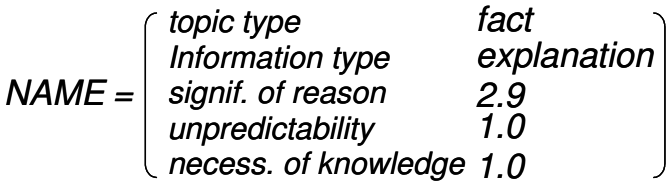


Figure 3.3: Example of topic feature structure

5. **Necessity of Knowledge and Experience:** This involves how important knowledge and experience are for the hearer to understand the topic. This feature takes a value between one to five.

The notation for representing a topic is shown as the matrix in Figure 3.2. Figure 3.3 illustrates an example about asking hearer's name (topic is NAME). In asking the partner's name (NAME), the value of *topic type* is *fact*, and the value of *information type* is *descriptive explanation* because the hearer must reply in his/her own words. The value of *significance of reason* is 2.9. This topic is highly predictable (the value of *unpredictability* is 1.0), and knowledge and experience are not so important to understand the topic (the value of *necessity of knowledge and experience* is 1.0). These values are based on the rating of twenty subjects.

3.1.3 Utterance Content Unit and Query Structure

In this section, we will propose the notion of the Utterance Content Unit (UC unit) as the basic constituent of query content. UC units are defined as the semantic and functional units of an utterance. The content of a query consists of a list of UC units. Representing a query as a list of UC units is very advantageous in generating natural language expressions because of its flexibility. The following two query expressions have the same utterance content but different surface expressions.
(a) "We can provide you with a push-type circuit or dial-type circuit. A push-type

circuit permits rapid connection while a dial-type circuit takes slightly longer."
(b) "We can provide you with a push-type circuit that permits rapid connection or a dial-type circuit that takes slightly longer."

The contents of these two expressions can be represented as a combination of the same two UC units: <Option Notification> and <Explanation about topic>. By using this framework, various expressions which have the same content can be generated. Moreover, people start speaking without determining all dialogue contents in advance, and add information or expressions when it seems reasonable. Representing a query as a list of UC units is suitable considering these characteristics of spoken expressions (Kato and Ishikawa 1992).

Based on the result of observing real conversations, we classified the UC units into two groups: kernel and auxiliary. Kernal UC units, for example <Yes/No interrogative> or <WH interrogative>, play a central role in the query. Auxiliary UC units are not the core, but supply additional information like <Explanation about topic>, or <Situation explanation>. Thus, the content of a query can be completely defined as a combination of Kernel UC unit(s) and Auxiliary UC unit(s).

Utterances that consist of only kernel UC unit(s), or only auxiliary UC unit(s) are possible. When asking the hearer's name, no auxiliary UC unit is added and only a kernel UC unit is uttered as in "Your name please." Utterances consisting of only auxiliary UC units are also acceptable because they can be regarded as "indirect speech acts". For example, verbalizing <Option Notification> as in "We can provide you with a push-type circuit or a dial-type circuit" can be interpreted as an indirect query. We observed many real-world conversations and extracted twenty kernel UC units and six auxiliary UC units. The kernel UC units are categorized based on the syntactical classification of interrogatives, such as <Yes/No interrogative>, <WH interrogative>, and <Coordinate interrogative>. <Request> is also treated as a kernel UC unit because queries can be expressed in request form, like "Please tell me..." Other kernel UC units are <Asking-permission> and <Confirmation>.

Auxiliary UC units are classified based on their pragmatic functions. The units are <Topic introduction>, <Option notification>, <Explanation about topic>, <Situation explanation>, <Restriction of the referents>, and <Asking about premise knowledge>. Auxiliary UC units are concerned with the felicitous condition of the utterance (Austin, 1962), and they are uttered to satisfy the preparatory condition in the speech act (Searle 1969). Figure 3.4 shows an example of analyzing a query. The first part of the utterance,"Next, we need your contact telephone number" is <Topic introduction> (auxiliary UC unit). The next part, "We may need to contact you" is <Situation explanation> (auxiliary UC unit), and the last part, "What number may I call?" is <WH interrogative> (kernel UC unit).

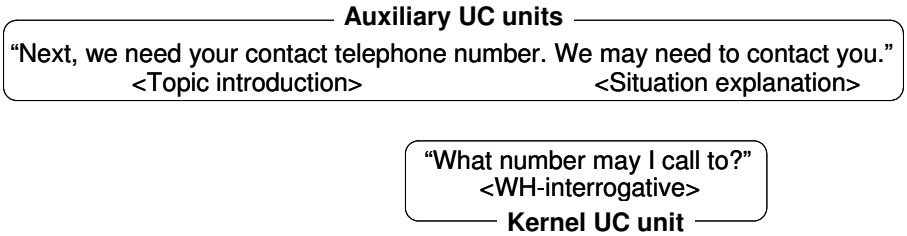


Figure 3.4: Structure of query

3.1.4 Utterance Content Planner (UCP)

This section explains the Utterance Content Planner (UCP) which selects the utterance content appropriate for a query according to its topic features. UCP input is a quintuple of topic feature values, and its output is a list of UC units as query content. Figure 3.5 shows the UCP architecture. This consists of a content selection part and a content construction part. In the content selection part, candidates for the kernel UC unit and the auxiliary UC unit are selected by using kernel selection rules and auxiliary selection rules, respectively. In the content construction part, some combinations of the kernel UC unit and auxiliary UC unit are produced as the complete query content.

Selecting Query Content

The input to the content selection part is a quintuple of topic feature values. They are used to select query content candidates as shown in the rules given below.

Kernel selection rules The kernel selection rules determine candidates for kernel UC units according to *topic type* and *information type* features. The kernel selection rules are shown in Table 3.1. In this table if the *topic type* is *will* and *information type* is *selecting option*, then <WH-interrogative> and <Coordinate interrogative> are selected as candidates for query content.

Auxiliary selection rules Auxiliary selection rules determine candidates for auxiliary UC units according to four topic features —*information type*, *significance of reason*, *unpredictability*, and *necessity of knowledge and experience*. As for information type, the following rule is applied:

> *If the value of information type is "selecting option type" then select <Option notification> as a candidate for the auxiliary UC unit.*

For the other three topic features, discriminant functions are applied that are defined for the UC units of <Explanation about topic>, <Situation Explanation>,

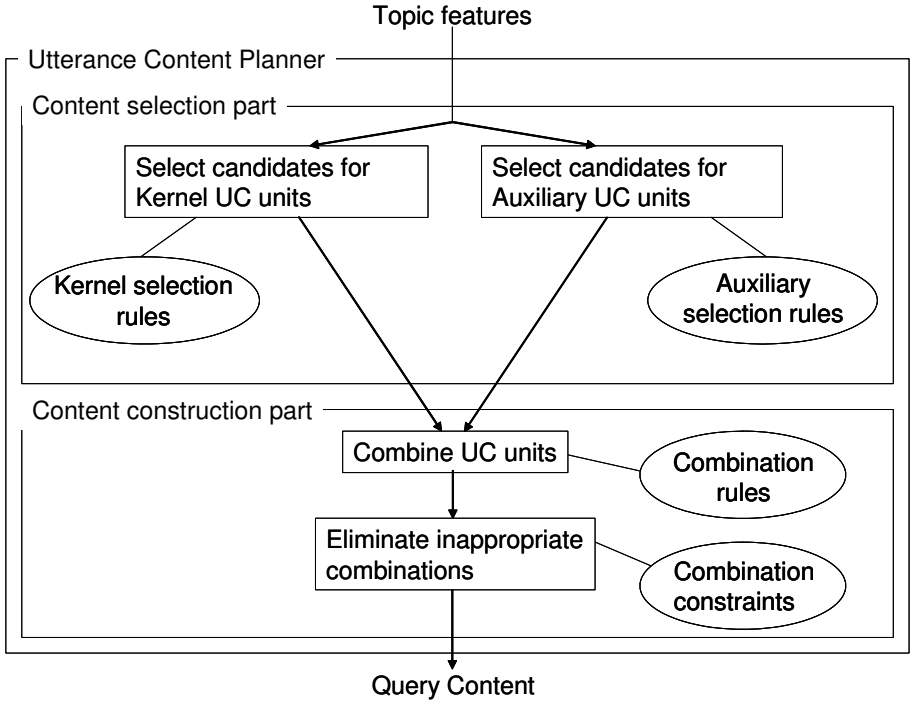


Figure 3.5: Utterance Content Planner

Table 3.1: Kernel Selection Rules

		Topic type	
		Fact	Will
Information type	Yes/No	Yes/No-interrogative WH-interrogative Confirmation	Yes/No-interrogative WH-interrogative Asking permission
	Selecting option	WH-interrogative Coordinate-interrogative	WH-interrogative Coordinate-interrogative
	Descriptive explanation	WH-interrogative Request	WH-interrogative

and <Asking for Premise Knowledge>. These functions take as variables three numerical topic features: *significance of reason*, *unpredictability*, and *necessity of knowledge and experience*. The function for each UC unit decides whether the UC unit is accepted as a candidate or not. If the value of the function exceeds a threshold, the UC unit is accepted as an auxiliary UC unit candidate. The discriminant function is represented as follows.

$$D(i) = \sum_{j=1}^{n} W_{ij} F_j(t)$$

i **:** Auxiliary UC unit

for i =<Explanation about topic>, <Situation Explanation>, <Asking for Premise Knowledge>.

j **:** topic features

for j = *significance of reason, unpredictability, necessity of knowledge and experience*

W_{ij} **:** weight of the topic feature j for auxiliary UC unit i

$F_j(t)$ **:** function that returns the value of topic feature j for topic t

$D(i)$ **:** disciminant function

τ_i **:** threshold for discriminant function $D(i)$

if $\tau_i \leq$ the value of $D(i)$ then accept the auxiliary UC unit i as a candidate.

if $\tau_i >$ the value of $D(i)$ then reject the auxiliary UC unit i.

The weights of topic features (W_{ij}) [1] and thresholds (τ_i) [2] are shown in Table 3.2.

The following is an example of discriminant function for <Explanation about topic>:

$$D(Explanation) = (-0.02)F_{reason}(t) + (0.202)F_{unpredic.}(t) + (-0.037)F_{knowledge}(t)$$

[1] As the weight of topic features, we adopted the partial regression coefficients β of the multiple regression equation that takes topic features values as predictor variables and estimates the frequency of the UC units as measured in actual conversations.

[2] Based on the general theory about discriminant functions, the threshold should be set midway between the mean values of $D(i)$ of the topics whose UC units i were actually observed and those of topics whose UC units i were not observed. This criterion for setting the threshold, however, makes the discriminant functions overestimate the candidates of auxiliary UC units. In order to improve the selection accuracy, we increase the threshold by 30%: that is, $\tau_i = 0.8$ (mean value of $D(i)$ of observed group) + 0.2 (mean value of $D(i)$ of non-observed group)

Table 3.2: Weight and Threshold in Discriminant Functions

		Topic features			
		Weigh (Wij)			Threshold (τ)
		signif. of reason	unpredic-tability	necess. of knowledge	
UC units	Explanation about topic	-0.020	0.202	-0.037	0.53 = D
	Situation explanation	0.169	0.084	-0.106	0.56 = D
	Asking for premise knowledge	0.249	0.066	-0.032	0.94 = D

.

Combining the Kernel and Auxiliary UC units

In the previous section, we introduced the rules that select the utterance content candidates. In this section, we explain the content construction part in UCP that combines the UC units to form the complete query content.

Some complex queries consist of two or more kernel UC units with several auxiliary UC units. We, however, consider hereafter only the simplest and most frequent queries. There are three types of such queries: one kernel UC unit plus one auxiliary UC unit, only one kernel UC unit, and only one auxiliary UC unit. In the content construction part, valid UC unit combinations are created based on combination rules and constraints.

Combination Rules of UC units

In the current construction part, the combinations of the kernel UC unit candidate and the auxiliary UC unit candidate are ranked by *combination rules* that utilize the importance of each auxiliary UC unit candidate. This ranking is needed because the UCP must be able to output the next most likely combination if the most likely combination constraints to be inappropriate. (These constraints are explained in the next section.) The combination rules are as follows:

(1) For each auxiliary UC unit candidate, the difference between the value of each discriminant function and its threshold τ_i is calculated. The candidate yielding the highest difference is taken as the first auxiliary candidate. The most preferred combination consists of the first auxiliary candidate and the kernel

UC unit. The second most preferred candidate is just the first auxiliary candidate. For example, if there are two candidates a and b, and $D(a)$-$\tau_a > D(b)$-τ_b (distance between a value of discriminant function *(Di)* and a threshold (τ_i) is greater in b than a), the ranking is (in order of decreasing preference) a plus kernel UC unit, auxiliary a only, b plus kernel UC unit, finally auxiliary b only. If there are more than one kernel UC unit candidates, all kernel UC units are considered to have equal ranking. As in the previous example, if the kernel UC unit candidates are x and y, the most preferred combinations are a plus x, and a plus y.

(2) If there is no auxiliary UC unit candidate, the most preferred combination consists of just the kernel UC unit.

Combination Constraints

After determining the most preferred combination, it is necessary to verify whether the combined utterance content is appropriate and informative as a whole. Inappropriate combinations are eliminated with heuristic combination constraints. Based on our analysis of actual Japanese conversations, we have extracted two constraints. We note that further analysis will yield more constraints.

(i) Kernel UC unit <Coordinate-interrogative> should not be combined with auxiliary UC unit <Option Notification>.

(ii) Kernel UC unit <Asking-permission> should not be combined with any auxiliary UC unit.

Constraint (i) prevents redundant utterances that refer to options twice. Constraint (ii) is quite conventional. In many cases, <Asking-permission> is not accompanied by any additional explanation, and is simply used to make the hearer confirm the speaker's utterance.

3.1.5 Example

In this section we show an example of producing a complete query content. The example involves the topic of asking whether the service of announcing the new telephone number is desired or not (ANNOUNCE SERVICE). This topic must be raised when a customer indicates to the telephone company that he/she is changing residence.

As explained in Section 3.1.2, the topic is characterized with a quintuple of topic features. The values were given as the rating of twenty subjects.

Topic type of ANNOUNCE SEVICE is *will* and *information type* is *yes/no*. As the result of applying these two values to the kernel selection rules in Table

Table 3.3: The Results of Applying Discriminant Functions to the Topic of "ANNOUNCE SERVICE"

	Threshold		
	Explanation about topic τ =0.53	Situation explanation τ =0.56	Asking for premise knowledge τ =0.94
Discriminant function values	0.54	0.54	0.90

3.1, <Yes/No-interrogative>, <WH-interrogative> and <Asking-permission> (such as "May I do something ...") are selected as the candidates for kernel UC units. Candidates for auxiliary UC units are decided by discriminant functions using the topic features of *significance of reason*: 3.0, *unpredictability*: 3.4, and *necessity of knowledge and experience*: 2.4. Table 3.3, shows the value of *D(i)* and threshold τ_i for each UC unit. The value of *D(i)* for <Explanation about topic> is 0.54. It exceeds the threshold 0.53, and thus this UC unit becomes an auxiliary UC unit candidate. <Situation explanation> and <Asking for premise knowledge> are not accepted because the values of the discriminant function *D(i)* for both of them do not exceed their respective thresholds.

The content construction part combines these candidates to yield the whole query content. The candidate for the auxiliary is <Explanation about topic> and kernel candidates are <WH-interrogative>, <Yes/No-interrogative>, and <Asking-permission>. The first step, according to combination rules, is to combine <Explanation about topic> with each kernel UC unit candidate. The combination constraints then eliminate the combination <Explanation about topic> and <Asking-permission> because this combination violates constraint(ii). Thus, the most preferred candidates are <Explanation about topic> + <WH-interrogative> and <Explanation about topic> + <Yes/No-interrogative>; the second alternative is <Explanation about topic> only. The surface expression of <Explanation about topic> + <Yes/No-interrogative> is, for example, "After we disconnect your current number, 012-345-6789, we can announce that you have moved and your new number is 987-654-3210 when someone calls the old number. Would you like this service?"

3.1.6 Evaluation of Utterance Content Planner (UCP)

This section describes the feasibility and effectiveness of UCP.

Observation data

We transcribed 160 telephone conversations about service reception. 120 conversations involved a telephone company reception service (changing residence), and 40 conversations were from a travel agency reception service (reserving a hotel or train tickets). These conversations contained 602 queries that involved 14 topics. After describing the 602 queries with UC units, we selected those queries (combination of UC units) that appeared more than three times as evaluation data. Thus, about 70% of the collected queries (416 queries) were used in the evaluation.

Correspondence to the real data

The topic feature values for 14 topics were specified by twenty subjects. As a result of applying these values to the UCP, 80% of the query contents observed in the dialogues were predicted (334 queries out of 416 were predicted by UCP). 45% of the combinations predicted by UCP were observed in the real data (49 combinations were generated by UCP and 22 of them appeared in the observed data). The high recall rate suggests that UCP can satisfactorily predict the query content in real-world conversations. The relatively low precision rate (42%) does not mean that UCP is inappropriate. The predicted combinations that failed to appear in the real data were reasonable and natural. We think that the precision rate will increase if the amount of observed data is increased.

Effectiveness of combination rules

The combination rules, which are generated for the service reception domain, allow UCP to generate the most appropriate and conventional query contents.

122 queries involving auxiliary UC units were predicted by UCP. 70% of them were the most preferred combinations, and 24% of them were ranked as the second preference. The order of candidates ranked by UCP accorded with the frequency in the real data.

3.2 Study 2: Factors for deciding utterance contents in instruction dialogues

In explanatory systems that give instructions, such as machine operation instructions, it is necessary to change the contents and presentation methods according to the history of interaction with a user and the model which represents the user's level of understanding. For example, when an explanation is given for a second time to a person who already knows the procedure quite well, the explanation is provided as a review of the instruction, and simpler explanations than the first time would

be effective. On the other hand, it is not effective to give the same instruction to a person who did not sufficiently understand the explanation the first time.

3.2.1 Previous work

Previous studies have proposed methods for deciding explanation content. (Mittal and Paris, 1993; Paris, 1991) proposed changing the explanation contents according to the user's knowledge level described in the user model. Other researchers proposed methods for selecting a detailed or simplified explanation strategy according to a parameter specifying the degree of detail in the explanation (Carletta, 1992; Moore, 1995; Moore and Swartout, 1989). However, in these methods, user models and parameters are pre-fixed. Strategies for deciding explanation contents are not changed dynamically by the system during the interaction. Therefore, it may happen that the system gives redundant explanations to well-understanding users.

As for the composition of explanation contents, previous studies on text generation proposed methods for generating a relatively short text as an answer to the user (Mittal and Paris, 1993; Moore, 1995; Paris, 1991). However, in instruction dialogues, a system should explain a long sequence of procedures. In such cases, it is preferable not to instruct all the procedures at once, but to explain the procedures interactively while ensuring the user's understanding through conversation.

Considering the characteristics of instruction dialogues, Cawsey (1990) proposed a mechanism that accepts the user's acknowledgement or clarification question every time the system finishes an utterance (i.e., end of the system's turn). Since, in her method, instruction dialogues are performed through short utterances exchange between the user and the system, it is possible to ensure the user's understanding interactively.

Note that floor management is an important issue in this approach because the system needs to accept the user's input in an appropriate timing. The system needs to be able to coordinate turns according to the state of the dialogue. In some cases, it would be more reliable to confirm the user's understanding for every single step of the procedure, whereas in reviewing the previous explanation for well-understanding users, it would be efficient to describe a few steps at a time without confirmation. However, Cawsey's work did not address such turn coordination issues.

Thus, this chapter first analyzes human instruction dialogues, and based on the results of the analysis, proposes heuristics for deciding utterance contents and turn boundaries according to the dialogue state. Then, we present our instruction dialogue system, where the heuristic rules are implemented.

In the following sections, first in Section 3.2.2, we report our analysis of human instruction dialogues, and show (1) how an instruction giver changes the utterance content according to the dialogue state, and (2) how s/he changes floor management

strategies in order to coordinate the conversation appropriately according to the dialogue state.

In Section 3.3, based on the results of analysis, we propose heuristics for deciding utterance contents as well as turns, then propose a planning mechanism using the heuristics. The instruction dialogue planning consists of utterance content planning and dialogue planning. To control the degree of carefulness of instruction, heuristics for utterance content planning are used in deciding whether an additional explanation should be provided or not. On the other hand, heuristics for deciding turn boundaries are used in dialogue planning to decide when the system should release the turn, and request a confirmation to the user. Finally, in Section 3.3.3, a sample dialogue with our instruction dialogue system is shown.

3.2.2 Analysis of Instruction Dialogues

Data

We collected 56 instruction dialogues between experts and novices an answering machine. The dialogues were collected under two conditions: telephone conversation situation (30 dialogues) and multimodal situation (26 dialogues) in which an instruction receiver can see an instruction giver's answering machine through a video monitor. In neither condition could the instruction giver monitor the receiver's operation. A preliminary study shows that the instruction utterances were not statistically different in these two situations though we had expected them to differ. Therefore, at least within the scope of this study, these dialogue data can be considered as homogeneous.

The average length of a dialogue is about 26 minutes, where a sequence of operations starting with the assembling of an answering machine is given to the receiver. In setting an answering message, first the instruction receiver practiced the operation by following the instruction giver's explanation. If the operation was successful, the receiver performed the operation again as a review. If the review operation was also successful, the task was completed. If the instruction receiver failed in the practice or review phase, s/he should perform the operation again.

Characteristics of instruction status

We characterize collected interactions using the following two factors.

The number of Task Trials (TT): The number of trials for an instruction receiver to perform a sequence of operations. If it is the first trial, the value is 0.

Learner's Level of Understanding (LLU): Although the value of this parameter should be continuous, in this study, we use two discrete values: "high" and

"low." If the instruction receiver succeeded in performing the task in the last session, the value is "high."

The state of the instruction dialogue is represented as a pair of values of these two factors.

Practice: The first time explanation for an instruction receiver. An instructor does not expect the receiver to know about the task. Thus, the factor value of TT is 0 and LLU is "low."

Review: An instructor has already explained the task at least once, and a learner successfully performed the task at the last trial. Thus, the value of TT is more than 1 and LLU is "high."

Re-explanation: A learner failed to perform the task at the last trial. Thus, TT is more than 1 and LLU is "low."

In the collected data, 56 Practice phases, 44 Review phases, and 20 Re-explanation phases were observed.

Analysis of utterance contents

In this section, we analyze the data to identify the determinant factors for selecting utterance contents and turn strategies. We analyze instructor's utterance contents using *Rhetorical Structure Theory* (RST) by Mann and Thompson (1987) (detailed description is in Section 2.2.1). In RST, the structure of a text is represented by the rhetorical relation between *nucleus* and *satellites*. The nucleus plays an essential role in expressing the speaker's purpose. Satellites provide supporting information to achieve the communicative goal. The nucleus also represents what a speaker intends the interlocutor to do, and satellites provide information that motivates the interlocutor to do it (Moser and Moore, 1996).

For example, if there are two utterances;
(a) "Push the response button 1."
(b) "Then, the button starts blinking."

Utterance (a) is the nucleus, and (b) is a satellite. Part (b) describes the result of accomplishing the behavior mentioned in (a), and supports the claim described in (a).

Employing this framework, we describe the structure of utterance content as a combination of the nucleus, which indicates the direction of an operation, and satellites, which provide additional information to the direction. Satellites observed in our data can be categorized as *Result* (result and meaning of accomplishing a behavior), *Elaboration* (describing the details and characteristics of behaviors or

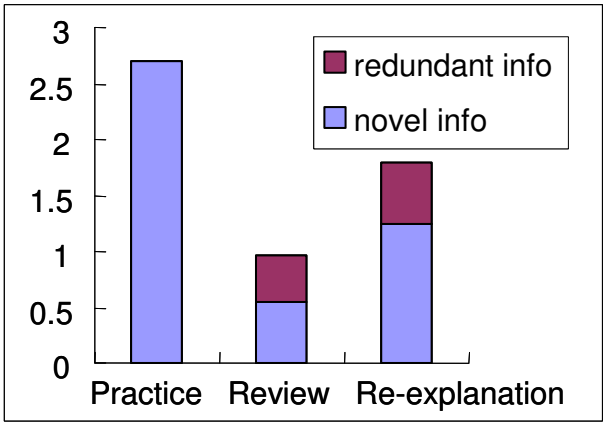


Figure 3.6: Account of additional information

objects), or *Condition* (condition required to perform a behavior relation,) which are defined by Mann and Thompson (1987) and Hovy (1993). Based on the discussion above, we investigated whether the number of satellites observed in the data is different depending on the state of the instruction dialogue.

Results of utterance content analysis

First, the frequencies of satellites for Practice, Review, and Re-explanation are shown in Figure 3.6. Satellites most frequently occurred in the Practice phase (2.70 per dialogue), and less frequently in Re-explanation (1.80). The frequency was lowest in the Review phase (0.97). The difference in frequency among these three phases is statistically significant ($F(2, 117)=19.344$, $p<0.0001$ [3]). This result suggests that the amount of additional information differs depending on the number of task trials and the learner's level of understanding.

To test the difference between the phases, Scheffe's test for multiple comparisons was conducted. As a result, the difference between Practice and Review was statistically significant ($p<0.001$), and the differences between Practice and Re-explanation, and between Review and Re-explanation were not statistically significant. Thus, the results were not sufficiently clear to determine whether the Re-explanation phase is closer to Review or Practice.

Then, we analyzed whether each satellite in the Review and Re-explanation phases included novel information that was first mentioned or redundant information which had already been mentioned in a previous dialogue (Figure 3.6). As

[3] ANOVA test was used for examining the difference of means. The result of the test is indicated as an F value. The numbers in parentheses indicate the degrees of freedom parameters. The significance degree of the result is indicated as a probability value (p). If p is less than 0.05, the result of the test is statistically significant.

a result, in the Review phase, the amount of redundant information was slightly larger than that of novel information. The frequencies of novel information and redundant information per instruction were 0.55 and 0.41, respectively. On the other hand, in Re-explanation, novel information was more frequently used than redundant information (frequencies were 1.25 and 0.55 respectively), and the difference was statistically significant ($t(38)=-2.252$ $p<0.05$ [4]). Therefore, in Re-explanation, the dialogue history affects the instruction. These results suggest that;

Discussion 1: At the first time explanation in which the learners' understanding levels are low, additional information is frequently used.

Discussion 2: In the Review phase, additional information is frequently omitted, and an instruction becomes simplified.

Discussion 3: As for Re-explanation, in giving additional information, novel information was preferred to redundant information which has already been mentioned. This result suggests that in Re-explanation, the dialogue history affects the selection of additional information.

Analysis of turn strategies

This section investigates how often an instruction giver releases a turn to a receiver to confirm the receiver's understanding, and what the determinant factors affecting turn strategies are. First, we define two types of turn strategies.

Keeping-turn strategy: Keeping the turn, and continuing onto the next utterance.

Releasing-turn strategy: Releasing the turn, and giving a partner (receiver) the chance to take a turn. This is frequently marked with an auxiliary verb or final particle, like "desu," or "tekudasai." In addition, this is frequently accompanied by a sentence-end intonation as well as a pause.

For example, in an utterance, "If you push the response button 1, the button starts blinking," the keeping-turn strategy is used in directing a core behavior (i.e., "push the response button 1 (`outou botan 1 wo osuto`)"). On the other hand, the releasing-turn strategy is used at the end of giving additional information (i.e., "the button starts blinking (`botan ga tenmetsu shimasu`)").

[4] T-test was used for examining the difference of means. The test result was indicated as a t value. The degree of freedom is shown in parentheses

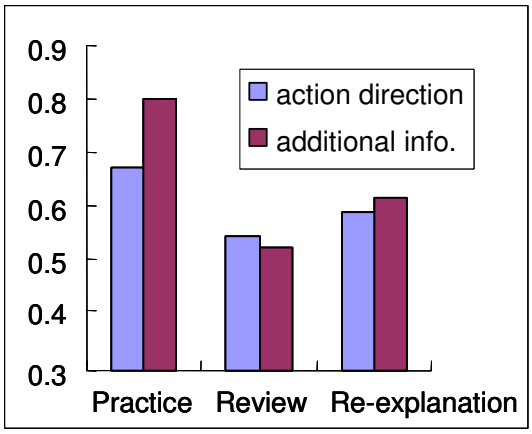


Figure 3.7: Percentage of turn release form

Results of turn strategy analysis

We investigated turn strategies from three aspects: the number of task trials, the learner's level of understanding, and the redundancy of the utterance (whether an utterance includes novel information or redundant information).

First, we investigated whether an action direction, which corresponds to the nucleus in a rhetorical relation, and additional information, which corresponds to a satellite, are expressed using the releasing-turn strategy or the keeping-turn strategy. Figure 3.7 shows the proportion of applying the releasing-turn strategy to an action direction and additional information in three types of instruction dialogue phases. In action directions, the releasing-turn strategy was most frequently used in the Practice phase (67%), least in Review (54%). This strategy was used 60% of the time in the Re-explanation phrase. Obviously, the keeping-turn strategy was used in the rest of the cases. The difference of proportion was statistically significant ($F(2,117)=6.297$, $p<0.003$). As a result of Scheffe's test for multiple comparisons, the difference between Practice and Review was statistically significant ($p<0.03$). The results were not sufficiently clear to determine whether the Re-explanation phase is closer to Review or Practice.

As for additional information which serves as a satellite in the rhetorical relation, [5] the releasing-turn strategy is most frequently used in the Practice phase (80%), next in the Re-explanation (62%), and least in Review (52%) (Figure 3.7). This

[5] As a satellite for a *Condition* the relation was almost always expressed as a subordinate clause placed before the main clause in a complex sentence, and the keeping-turn strategy was always preferred regardless of the state of dialogue. Thus, we investigated additional information except for *Condition*.

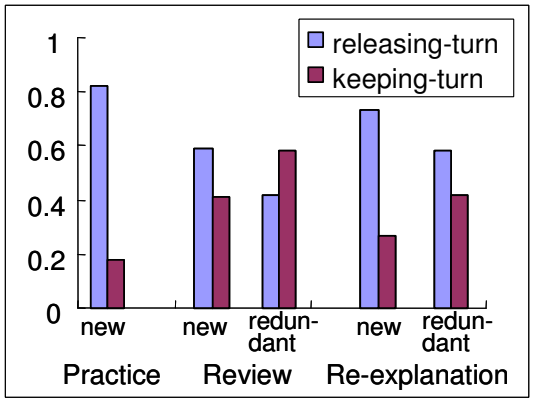


Figure 3.8: Turn strategy for novel/redundant information

result indicates that the turn strategy for both action directions and additional information is determined by two factors: the number of task trials and the learner's level of understanding.

We also investigated whether the redundancy of utterance is a determinant factor of turn strategy. We analyzed additional information for *Result* relations for which we obtained sufficient data to conduct statistical analysis. Proportions of turn strategies for novel and redundant information mentioned in the *Result* relation are shown in Figure 3.8. In Practice, because of the first time explanation, all the information is novel, and 82% of the cases were expressed using the releasing-turn strategy. This result is consistent with that obtained in the last paragraph.

As for novel information mentioned in Re-explanation, the releasing-turn strategy was used in 73% of the cases, and the keeping-turn strategy was applied to the rest of the cases (27%). This indicates that the releasing-turn strategy was definitely preferred. We did not find any clear preference for the turn strategy in the Review phase.

These results suggest that, specifically in Re-explanation, redundancy of utterance affects turn strategies. The turn strategy for novel information changed according to the state of dialogue, whereas that for redundant information was not very different between Practice and Re-explanation.

Walker (1992) claimed that redundant utterances, which theoretically violate the maxim of quantity in Grice's Conversational Maxims (Grice, 1975), contribute to establishing mutual belief in a real conversation. We found that such redundant information is presented in a different way from novel information. Now, we can summarize our discussions as follows.

Discussion 4: In the Practice phase, the instruction givers give action directions and additional information through relatively short utterances, and frequently

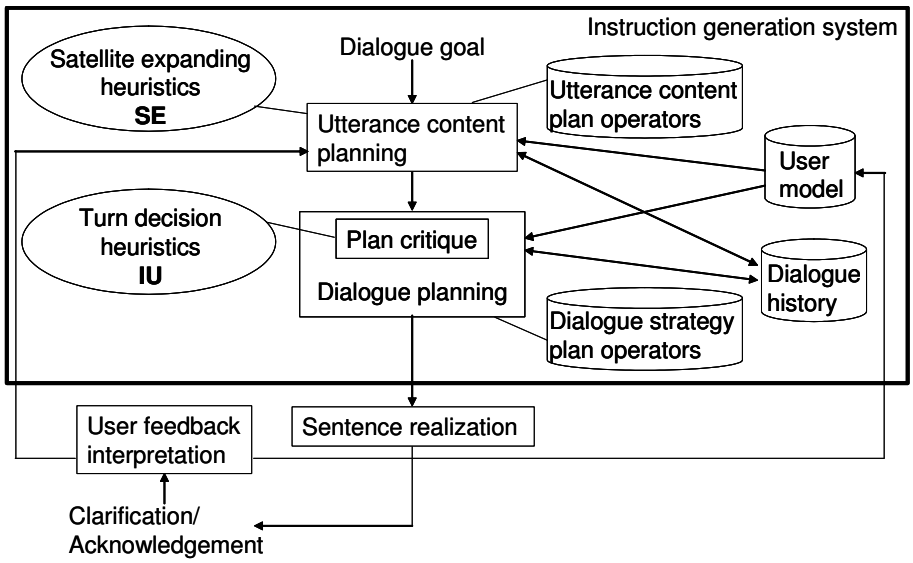


Figure 3.9: The architecture of the instruction generation system

confirm the instruction receiver's understanding.

Discussion 5: On the contrary, in the Review phase, the instruction givers try to shorten and simplify a conversation by avoiding redundant additional information, and conveying a larger chunk of information in a turn.

Discussion 6: In the Re-explanation phase, the instruction givers try to correct the receiver's misunderstanding by giving a different explanation from the previous one, and splitting the explanation into smaller chunks.

3.3 Planning instruction dialogues

Based on the empirical study reported in the previous section, this section describes an implementation of an instruction dialogue system. First, the overall process of this system is described. Then, we propose heuristics for selecting the utterance content and turn strategy based on the results of our empirical study.

3.3.1 Planning mechanism

The system architecture of our instruction dialogue system is shown in Figure 3.9. [6] When an explanation goal (e.g., instruct how to record an answering message) is input to the system, first the utterance content planner determines the utterance

[6] In this study, modules enclosed by a thick line are implemented.

HEADER	((Instruct S H ?act)
EFFECT	((BMB S H (Goal H (Done H ?act)))
	(BMB S H (Know H (How-to-do ?act))))
CONSTRAINTS	((Goal S (Done H ?act))
	(Step ?act ?g-goal))
NUCLEUS	(Command S H (Done H ?act))
SATELLITES	(((Persuade S H (Done H ?act)) *optional*)
	((Achieve S (BMB S H (Competent H (Done H ?act)))) *optional*))

Figure 3.10: Example utterance content plan operator

content by expanding the plan using utterance content plan operators. We employ a hierarchical planning mechanism and incrementally expand the plan using a depth-first search. This method has been proposed in (Moore and Paris, 1989; Cawsey, 1992), and is suitable for generating an instruction dialogue for a long sequence of operations. This is because, in this method, the plan can be changed dynamically according to the user's response.

The design of a plan operator is similar to that proposed by (Moore, 1995; Moore and Paris, 1989). An operator consists of *header*, *effect* by executing a plan, *constraint* on executing a plan, *nucleus* as a mandatory sub-goal, and *satellite* as an optional sub-goal supporting a nucleus. An example of a plan operator is shown in Figure 3.10. This operator is used for instructing an action. There are two effects in this operator: (1) the goal of the action is mutually believed; (2) the hearer becomes to know how to do the action.[7] The constraints on executing the operator are that the speaker has the goal of making the hearer perform the action, and the action contributes to accomplish the global goal. This operator has a nucleus (Command S H (Done H ?act)), which commands an action, and has two satellites: (Persuade S H (Done H ?act)) and (Achieve S (BMB S H (Competent H (Done H ?act))))). The first satellite is to persuade the hearer to perform the action. The second one is to make the hearer become competent in performing the action. Some satellites are mandatory and others are optional. Both satellites in Figure 3.10 are optional (marked with *).

As a result of utterance content planning, primitive propositions are output to the dialogue planning module. In dialogue planning (details will be described in Section 3.3.2), first the plan critique module determines the turn strategy: whether to keep or release a turn. Then, the planning mechanism, which is similar to the content planner, decides how to express the utterance content to construct a turn. Figure 3.11 shows examples of dialogue strategy plan operators. Dialogue strategy operators consist of *header*, *constraints*, and *subgoals*.[8]

[7](BMB S H P) is a simplified description of (BEL S (MB S H P)), which means that S believes that a proposition P is shared between S and H.

[8]Terms in discourse analysis, such as *exchange*, *move*, and *act*, are used in order to describe

```
HEADER          (Instruct-exchange ?contents)
CONSTRAINTS     ((Content-type ?contents DECLARE))
SUBGOALS        ((S-inform-move ?contents)
                 (U-reply-move ?contents))

HEADER          (Instruct-exchange ?contents)
CONSTRAINTS     ((Content-type ?contents COMMAND))
SUBGOALS        ((S-Request-move ?contents)
                 (U-answer-move move ?contents))
```

Figure 3.11: Example dialogue strategy plan operators

The output from the dialogue planner is sent to the sentence generator to generate a surface expression in Japanese, and the final output is produced through a text-to-speech engine. When the user acknowledges the system's instruction utterance, planning for the next utterance starts.

Subgoals are maintained in the agenda shared by the utterance content planning module and the dialogue planning module. This idea is similar to Cawsey (1992). The system picks up the subgoal to be expanded next, finds a plan operator to accomplish the subgoal, then puts the subgoals produced by expanding the selected operator on the top of the agenda. This process allows the plan to be expanded in a depth-first manner. As the agenda allows the system to change the plan flexibly, it is useful for generating instruction dialogue in which the state of a dialogue dynamically changes.

3.3.2 Heuristics for plan selection

To generate appropriate instruction utterances according to the dialogue context, the planning mechanism uses two types of heuristics: (1) satellite expansion heuristics, which are used in content planning to decide whether additional information should be added or not, and (2) turn-taking decision heuristics, which are used in the plan critique module to form a turn unit from multiple utterance contents. These heuristics are based on the results of analyzing instruction dialogues in Section 3.2.2. They are used in deciding utterance contents and turn strategies according to the values of *the number of task trials*, *the learner's level of understanding*, and *redundancy of utterance*.

---

dialogue strategy plan operators (Cawsey, 1992).

Dialogue history and user model

First, we define the dialogue history and the user model, both of which are referred by the heuristics.

The dialogue history is used for obtaining information about *the number of task trials*. The dialogue history stores the plan tree for the current dialogue as well as those for previous dialogues. Therefore, by checking the dialogue history, we can find which goals have already been accomplished. Moreover, by looking at the leaves of plans in the previous dialogues, which indicate utterances already generated by the system, we can obtain information about redundancy of the utterance.

In the user model, the user's knowledge, belief, and goals are described. The user model is updated every time utterances are exchanged. For example, whenever the user acknowledges the system's utterance, the effects of the utterance are added to the user model. If the user failed to perform the task, all the beliefs related to performing the task are canceled. Thus, the learner's level of understanding is judged by looking at the user model.

Heuristics for expanding satellites

As described in Section 3.3.1, utterance contents are selected by expanding a plan using utterance content plan operators in utterance content planning. Moore (Moore, 1995; Moore and Paris, 1989) implemented two types of plan expansion policies: not expanding a satellite at all, and expanding satellites as much as possible. [9] However, in their implementation, the policy is pre-fixed before starting a conversation, and the policy cannot be changed during the conversation. When generated explanations are short, this method does not cause a problem. However, in generating instructions with a long sequence of interaction, it is necessary to change the expansion policies flexibly. In some cases, additional information should be mentioned as much as possible. In other cases, additional information should be omitted as much as possible to simplify the instruction.

We employ a method that changes the policy for expanding satellites at any time in content planning. For this purpose, we propose the following three heuristics using the dialogue history and the user model. These heuristics are applied in selecting plan operators, and used for selecting the most appropriate operator among the candidates according to the state of the dialogue. **SE1** was derived from **Discussion 1**, **SE2** was from **Discussion 2**, and **SE3** was from **Discussion 3**.

<Satellites expanding heuristics>

[9] In expanding satellites as much as possible, by looking at the user model and the dialogue history, the system expands satellites which do not overlap with the user's knowledge.

SE1 In each candidate plan operator, if the nucleus of the operator is not registered in the dialogue history and the effects of the operator are not registered in the user model, it is preferable to select a candidate that has at least one satellite.

SE2 In each candidate plan operator, regardless of whether the nucleus of the operator is registered in a dialogue history or not, if the effects of the operator have already been registered in the user model (or they are presumed to be the user's pre-existing knowledge), it is preferable to select a candidate that does not have any satellites.

SE3 In each candidate plan operator, if the nucleus of the operator has already been registered in the dialogue history, but the effects of the operator are not registered in the user model, it is preferable to select a candidate that contains satellites contributing to the correction of misunderstanding.

In Practice as the first time explanation, **SE1** is applied, and plan operators containing satellites are preferred. In the Review phase, **SE2** is preferred, and operators without satellites are preferred. In Re-explanation, **SE3** is more preferably applied, and satellites are expanded in order to correct misunderstanding. Since, in the current system, the cause of misunderstanding cannot be identified, plan operators containing satellites which have not been mentioned yet are more preferable. As a result, novel additional information is preferably mentioned. Note that this case has been actually observed in our data analysis, where novel additional information is preferred to redundant additional information in the Re-explanation phase.

Moreover, when the system needs to clarify a part of the sequence of operations, to correct partial misunderstanding, **SE3** is also applied. Then, to resume the dialogue after finishing the re-explanation, **SE2** is applied, and previously mentioned explanations are given again in simple and short utterances. This strategy is called "back-on-track repletion," and, in our data, we actually found some real cases that used this resuming strategy. This strategy is also useful in recovering from interruption, where the user's attention needs to be drawn back to the original point (Mooney, Carberry, and McCoy, 1991).

Heuristics for deciding a turn

In some cases of instruction dialogues, it is better to instruct operations step by step while confirming the user's understanding. In other cases, it is preferable to give instructions for multiple steps at one time. Cawsey (1992) proposed a method for generating an instruction dialogue by planning a dialogue strategy, whereas she did not propose how to coordinate turns according to the state of dialogue or how to change the amount of information conveyed at one time. Thus, in order to coordinate

turn-taking according to the state of dialogue, this section proposes heuristics for deciding turns.

In dialogue planning, first, the plan critique module is triggered in order to construct a turn. Applying the following heuristics to individual utterance content produced by the utterance content planner, this module judges whether a given utterance content should be combined with the proceeding content to construct a turn. Similar to the satellites expanding heuristics, these heuristics refer to the dialogue history and the user model. In addition, these are based on the results of our empirical study reported in Section 3.2.2, and derived directly from **Discussion 4** and **6** (for **IU1**), and **Discussion 5** (for **IU2**).

<Turn decision heuristics>

IU1 For a given utterance content, if neither the utterance content nor the effects of uttering it have been registered in the dialogue history and the user model respectively, it is preferable to release a turn to the user by taking the releasing-turn strategy.

IU2 For a given utterance content, if it has been registered in the dialogue history and the effects of the utterance are registered in the user model, it is preferable to keep the turn by taking the keeping-turn strategy.

When **IU2** is selected, the utterance content is temporarily saved in the *utterance contents list*. On the other hand, when **IU1** is selected, the utterance content is added to the utterance contents list. Then, the list is put onto the top of the agenda as a turn, and the system goes onto the planning for deciding an interaction type (more details will be provided in the next section).

In Practice and Re-explanation, **IU1** is preferred to be applied, and individual utterance content is generated using the releasing-turn strategy. On the other hand, in the Review phase, **IU2** is preferred to be applied, and utterance contents are presented using the keeping-turn strategy. Even in the Review phase, when a directed action requires a user to take time, **IU1** is applied in order to give the user time to accomplish the operation. This constraint is also useful to avoid generating all the instructions in one turn. Moreover, **IU2** is applied in resuming from a repair dialogue.

Deciding interaction type

After grouping utterance contents as a turn, the plan critique module continues planning by applying dialogue strategy plan operators to the collection of utterance contents. This step specifies the type of exchange for the turn. The result is the system final output.

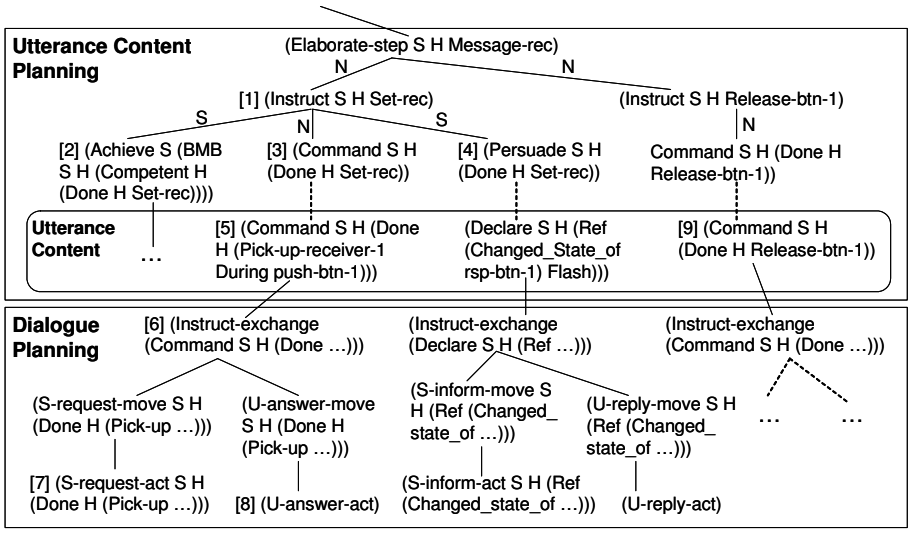


Figure 3.12: A partial instruction dialogue plan for the Practice phase

For example, <Operator 1> in Figure 3.11 shows that an exchange between a system and a user consists of two moves: S-inform-move and U-reply-move. If the last utterance content in a turn, which notifies the speaker's turn release, is a declaration, <Operator 1> is selected. If the last utterance content is a request, <Operator 2> is selected. In <Operator 1>, the system requires the user's response to the system's inform move. In <Operator 2>, it requires the user's answer to the system's request move.

3.3.3 Example of instruction dialogue

The whole instruction planning mechanism described above is integrated into a multimodal dialogue system (Kato et al., 1996a) that explains the initial setting of an answering machine using speech, text, graphics, and pointing, etc, and works as a speech guidance generation component in the entire system. The output of the instruction planning mechanism is sent to the sentence realization component where the utterance content is transformed into a Japanese sentence. Then, the linguistic expression is converted into speech sound through a text-to-speech engine. Finally, the speech sound, graphics, and pointing animations are presented to the user in a synchronized manner. In this system, the user can respond to the system by selecting one of the items in a menu. [10]

[10]If a user selects "eh?," the system considers that the user did not understand the most recent instruction. In this case, the system re-plans the most recent goal, and generates a re-explanation of it. If a question about what has already been explained is selected, the system generates a re-explanation about the operation and then comes back to the interrupted point using the resuming

<Practice>
S: Pick up the receiver while pushing the response button 1.
U: Yes.
S: The red response lamp one starts blinking.
U: Yes.
S: Release the response button 1.
U: Yes.
S: Speak the answering message.

<Review>
S: Pick up the receiver while pushing the response button 1, release the response button, and speak the answering message.
U: Yes.

<Re-explanation>
S: Pick up the receiver while pushing the response button 1.
U: Yes.
S: Then, the machine status changes to recording.
U: Yes.
S: Release the response button 1.
U: Yes.
S: Speak the answering message.

Figure 3.13: Example dialogues between the system and a user

A plan for a dialogue in the Practice phase is shown in Figure 3.12. In utterance content planning, the instruction content is determined by expanding the plan, and the preference in selecting a plan operator is specified in satellites expanding heuristics (**SE1**–**SE3**). In this example, as a Practice phase instruction is being planned, **SE1** is applied and operators including any satellites are preferred. In [1] (Instruct S H Set-rec), two satellites are expanded ([2] (Achieve S (BMB S H (Competent H (Done H Set-rec)))) to identify the response button 1, and [4] (Persuade S H (Done H Set-rec)) which describes the operation result. Utterance contents generated as a result of expanding this plan are sent to the dialogue planner one by one.

In dialogue planning, first the plan critique module applies turn decision heuristics (**IU1 IU2**) to each utterance content, and constructs a set of utterance contents conveyed in one turn. In this example, **IU1** is applied to all the utterance contents,

strategy. A dialogue with the system starts with the Practice phase, and after a sequence of instruction is performed, a dialog pops up on the display to ask the user whether the operation was successful. If the user selects "Yes',' a Review explanation is generated. If the user selects "No," a Re-explanation is generated as the second time explanation.

and the releasing-turn strategy is selected. An output from the plan critique module is similar to [6] (Instruct-exchange (Command S H (Done H (Pick-up-receiver-1 During Push-btn-1)))). Then, the dialogue planner applies dialogue strategy plan operators to these utterance contents, and the dialogue plan is expanded. For example, [6] (Command S H (Done H (Pick-up-receiver-1 During Push-btn-1))) is expanded to an exchange consisting of an action direction by the system: [7](S-request-act S H (Done H (Pick-up-receiver-1 During Push-btn-1))), and the user's response to the command: [8] (U-answer-act). The first part of the Figure 3.13 (<Practice>) shows an example dialogue for this part. S indicates the system's turn and U indicates the user's turn.

As for an explanation in the Review phase, **SE2** is applied in utterance content planning, and operators without satellites are preferred. For example, two satellites ([2], [4]) expanded in Figure 3.12 are omitted in a review explanation. In dialogue planning, in most of the cases, the keeping-turn strategy is selected by applying the **IU2**, and the content is temporarily saved in the utterance contents list. In this example, the keeping-turn strategy is applied to two utterance contents: [5] (Command S H (Done H (Pick-up-receiver-1 During Push-btn-1))) and [9] (Command S H (Done H Release-btn-1)). Then, right after (Command S H (Done H Speak-message)) (which is not in the Figure 3.12) is processed and the releasing-turn strategy is selected, so all the utterance contents in the list are put together as a turn. Then, the system continues onto the dialogue planning, and the dialogue strategy is determined using dialogue strategy plan operators. As a result, the dialogue shown in Figure 3.13 <Review> is generated.

Finally, an example of the re-explanation interaction is shown at the bottom of Figure 3.13, <Re-explanation>. In giving a re-explanation to a user who does not understand well about a given operation, **SE3** is used in utterance content planning. Note that, at this step, plan operators that have satellites but have not been expanded yet are preferred (see Section 3.3.2). In the example of the Practice phase, "the red response lamp one starts blinking" is mentioned as additional information. On the contrary, in Re-explanation, "then, the machine status changes to recording" is mentioned as additional information instead. In dialogue planning, **IU1** is applied and turn-releasing strategy is preferred.

3.4 Summary

In this chapter, first we conducted an empirical study and found that utterance contents and dialogue strategies are different according to the state of the instruction dialogue: Practice, Review, and Re-explanation. Then, based on the empirical results, we proposed heuristics for generating instruction dialogues and implemented them into a planning mechanism. During the process of planning the contents of an

instruction dialogue, the *satellites expanding heuristics* decides whether additional information should be mentioned or not. This mechanism allows the system to change the content of the explanation according to the instruction state: successful or problematic. Moreover, *turn decision heuristics* contribute to calculating an appropriate amount of information conveyed in one turn. These heuristics allow the instruction generation system to generate a variety of instruction dialogues.

As a future direction, it would be important to consider other determinant factors for deciding utterance content and a turn. For example, rhetorical relations between multiple utterance contents should be considered in constructing an instruction dialogue (Linden, 1994; Linden and Martin, 1995). As for the size of a turn, limitations of the human short term memory would be a useful concept to prevent infinite connection of utterance contents (Walker and Rambow, 1994).

Chapter 4

Generating Multimodal Instruction Dialogues

By extending the previous chapter, this chapter describes a generation mechanism of instruction dialogue in a virtual environment. First, we discuss what is necessary for generating multimodal explanatory dialogue in addition to the factors that we have already pointed out in the previous chapter, such as the discourse history, the user model, and the domain knowledge (topic characteristics).

According to the discussion, we focus on a user's view as a factor determining instruction dialogue, and propose a method for altering the instruction dialogue to match the user's view in a virtual environment. We illustrate the method with the system MID-3D, which interactively instructs the user on dismantling some parts of a car. First, in order to change the content of the instruction dialogue to match the user's view, we extend the refinement-driven planning algorithm by using the user's view as a plan constraint. Second, to manage the dialogue smoothly, the system keeps track of the user's viewpoint as part of the dialogue state information and uses this for coping with interruptive subdialogues. These mechanisms enable MID-3D to set instruction dialogues in an incremental way; it takes account of the user's view even when it changes frequently.

This chapter is organized as follows. The next section provides motivation and overview of this chapter. In Section 4.2, we define the problems specific to 3D multimodal dialogue generation. Section 4.3 describes related works. In Section 4.4, we propose the MID-3D architecture. Sections 4.5 and 4.6 describe the content planning mechanism and the dialogue management mechanism, and show how they dynamically decide coherent instructions, and control mixed-initiative dialogues considering the user's view. We also show a sample dialogue in Section 4.7.

4.1 Overview

In a 3D virtual environment, we can freely walk through the virtual space and view three dimensional objects from various angles. A multimodal dialogue system for such a virtual environment should aim to realize conversations which are performed in the real world. It would also be very useful for education, where it is necessary to learn in near real-life situations.

One of the most significant characteristics of 3D virtual environments is that the user can select her/his own view from which to observe the virtual world. Thus, the multimodal instruction dialogue system should be able to set the course of the dialogue by considering the user's current view. However, previous works on multimodal presentation generation and instruction dialogue generation (Wahlster et al., 1993; Moore, 1995; Cawsey, 1992) do not achieve this goal because they were not designed to handle dialogues performed in 3D virtual environments.

This chapter proposes a method that ensures that the course of the dialogue matches the user's view in the virtual environment. More specifically, we focus on (1) how to select the contents of the dialogue since it is essential that the instruction dialogue system form a sequence of dialogue contents that is coherent and comprehensible, and (2) how to control mixed-initiative instruction dialogues smoothly, especially how to manage interruptive subdialogues. These two problems basically determine the course of the dialogue.

First, in order to decide the appropriate content, we propose a content selection mechanism based on plan-based multimodal presentation generation (André and Rist, 1993; Wahlster et al., 1993). We extend this algorithm by using the user's view as a constraint in expanding the plan. In addition, by employing the incremental planning algorithm, the system can adjust the content to match the user's view during on-going conversations.

Second, in order to manage interruptive subdialogues, we propose a dialogue management mechanism that takes account of the user's view. This mechanism maintains the user's viewpoint as a dialogue state in addition to intentional and linguistic context (Rich and Sidner, 1998). It maintains the dialogue state as a focus stack of discourse segments and updates it at each turn. Thus, it can track the viewpoint information in an on-going dialogue. By using this viewpoint information in resuming the dialogue after an interruptive subdialogue, the dialogue management mechanism returns the user's viewpoint to that of the interrupted segment.

These two mechanisms work as a core dialogue engine in MID-3D (Multimodal Instruction Dialogue system for 3D virtual environments). They make it possible to set the instruction dialogue in an incremental way while considering the user's view. They also enable MID-3D to create coherent and mixed-initiative dialogues in virtual environments.

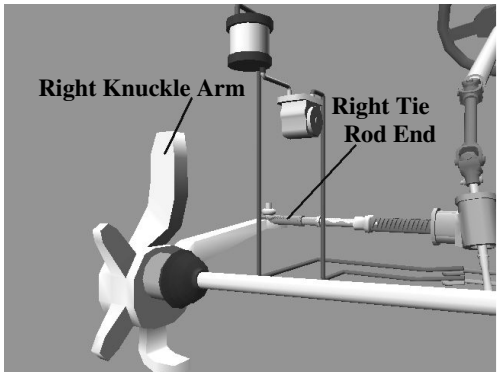


Figure 4.1: Right angle

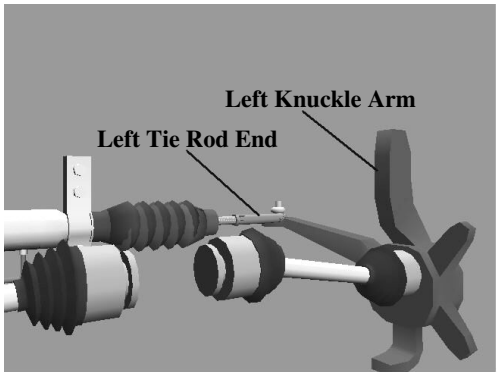


Figure 4.2: Left angle

4.2 Problems

In a virtual environment, the user can freely move around the world and select her/his own view. The system cannot predict where the user will stand and what s/he observes in the virtual environment. This section describes two types of problems in generating instruction dialogues for such virtual environments. They are caused by mismatches between the user's viewpoint and the state of the dialogue.

First, the system should check whether the user's view matches the focus of the next exchange when the system tries to change communicative goals. If a mismatch occurs, the system should choose the instruction dialogue content according to the user's view. Figure 4.1 and 4.2 are examples of observing a car's front suspension from different points of view. In Figure 4.1, the right side of the steering system can be seen, while Figure 4.2 shows the left side. If the system is not aware of the user's view, the system may talk about the left tie rod end even though the user's view remains the right side (Figure 4.1). In such a case, the system should change its description or ask the user to change her/his view to the left side view (Figure

4.2) and recommence its instruction about this part. Therefore, the system should be able to change the content of the dialogue according to the user's view. In order to accomplish this, the system should have a content selection mechanism which incrementally decides the content while checking the user's current view.

Second, there could be a case in which the user changes the topic as well as the viewpoint as interrupting the system's instruction. In such a case, the dialogue system should keep track of the user's viewpoint as a part of the dialogue state and return to that viewpoint when resuming the dialogue after the interrupting subdialogue. Suppose that while the system is explaining the right tie rod end, the user initially looks at the right side (Figure 4.1) but then shifts her/his view to the left (Figure 4.2) and asks about the left knuckle arm. After finishing a subdialogue about this arm, the system tries to return to the dialogue about the interrupted topic. At this time, if the system resumed the dialogue using the current view (Figure 4.2), the view and the instruction would become mismatched. When resuming the interrupted dialogue, it would be less confusing to the user if the system returned to the user's prior viewpoint rather than selecting a new one. The user may be confused if the dialogue is resumed but the observed state looks different.

We address the above problems. In order to cope with the first problem, we present a content selection mechanism that incrementally expands the content plan of a multimodal dialogue while checking the user's view. To solve the second problem, we present a dialogue management mechanism that keeps track of the user's viewpoint as a part of the dialogue context and uses this information in resuming the dialogue after interruptive subdialogues.

4.3 Previous work

There are many multimodal systems, such as multimedia presentation systems and animated agents (Maybury, 1993; Lester et al., 1997; Bares and Lester, 1997; Stone and Lester, 1996; Towns, Callaway, and Lester, 1998), all of which use 3D graphics and 3D animations. In some of them (Maybury, 1993; Wahlster et al., 1993; Towns, Callaway, and Lester, 1998), planning is used in generating multimodal presentations including graphics and animations. They are similar to MID-3D in that they use planning mechanisms in content planning. However, in presentation systems, unlike dialogue systems, the user just watches the presentation without changing her/his view. Therefore, these studies are not concerned with changing the content of the discourse to match the user's view.

In some studies of dialogue management (Rich and Sidner, 1998; Stent et al., 1999), the state of the dialogue is represented using Grosz and Sidner's framework (Grosz and Sidner, 1986). We also adopt this theory in our dialogue management mechanism. However, they do not keep track of the user's viewpoint information as

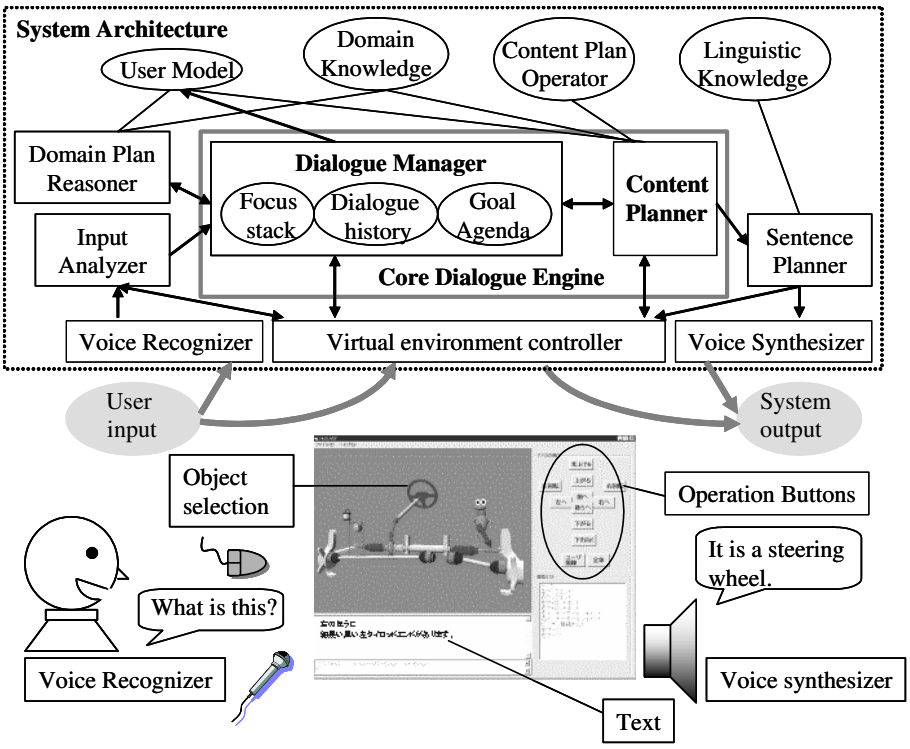


Figure 4.3: The system architecture

a part of the dialogue state because they were not concerned with dialogue management in virtual environments.

Studies on pedagogical agents have goals closer to ours. In (Rickel and Johnson, 1999), a pedagogical agent demonstrates the sequential operation of complex machinery and answers some follow up questions from the student. Lester, Stone, and Stelling (1999) proposes a life-like pedagogical agent that supports problem-solving activities. Although these studies are concerned with building interactive learning environments using natural language, they do not discuss how to decide the course of on-going instruction dialogues in an incremental and coherent way.

4.4 MID-3D System Architecture

This section describes the architecture of MID-3D. This system instructs users how to dismantle the steering system of a car. The system steps through the procedure. The user can interrupt the system's instructions at any time. Figure 4.3 shows the architecture and a snapshot of the system. The 3D virtual environment is viewed through an application window. A 3D model of a part of the car is provided and a frog-like character is used as the pedagogical agent (Johnson, Rickel, and Lester, 2000b). The user herself/himself can also appear in the virtual environment as an

avatar. The buttons to the right of the 3D screen are operation buttons for changing the viewpoint. By using these buttons, the user can freely change her/his viewpoint at any time.

This system consists of five main modules: Input Analyzer, Domain Plan Reasoner, Content Planner (CP), Sentence Planner, Dialogue Manager (DM), and Virtual Environment Controller.

First of all, the user's inputs are interpreted through the Input Analyzer. It receives strings of characters from the voice recognizer and the user's inputs from the Virtual Environment Controller. It interprets these inputs, transforms them into a semantic representation, and sends them to the DM.

The DM, working as a dialogue management mechanism, keeps track of the dialogue context including the user's view and decides the next goal (or action) of the system. Upon receiving an input from the user through the Input Analyzer, the DM sends it to the Domain Plan Reasoner (DPR) to get discourse goals for responding to the input. For example, if the user requests some instruction, the DPR decides the sequence of steps that realizes the procedure by referring to domain knowledge. The DM then adds the discourse goals to the goal agenda. If the user does not submit a new topic, the DM continues to expand the instruction plan by sending a goal in the goal agenda to the CP. Details of the DM are given in Section 4.6.

After the goal is sent to the CP, it decides the appropriate contents of instruction dialogue by employing a refinement-driven hierarchical linear planning technique. When it receives a goal from the DM, it expands the goal and returns its subgoal to the DM. By repeating this process, the dialogue contents are gradually specified. Therefore, the CP provides the scenario for the instruction based on the control provided by the DM. Details of the CP are provided in Section 4.5.

The Sentence Planner generates surface linguistic expressions coordinated with action (Kato et al., 1996a). The linguistic expressions are output through a text-to-speech engine. Actions are realized through the Virtual Environment Controller as 3D animation.

For the Virtual Environment Controller, we use HyCLASS (Kawanobe et al., 1998), which is a 3D simulation-based environment for educational activities. Several APIs are provided for controlling HyCLASS. By using these interfaces, the CP and the DM can discern the user's view and issue an action command in order to change the virtual environment. When HyCLASS receives an action command, it interprets the command and renders the 3D animation corresponding to the action in real time.

```
<Operator 1>
 (:Header          (Instruct-act S H ?act MM)
 :Effect           (BMB S H (Goal H (Done H ?act)))
 :Constraints      ((KB (Obj ?act ?object))
                   (Visible-p (Visible ?object t)))
 :Main-Acts        ((Look S H)
                    (Request S H (Try H (action ?act)) NO-SYNC MM))
 :Subsidiary-Acts ((Describe- act S H ?act MM)
                    (Reset S (action ?act))))

<Operator 2>
 (:Header          (Instruct-act S H ?act MM)
 :Effect           (BMB S H (Goal H (Done H ?act)))
 :Constraints      ((KB (Obj ?act ?object))
                    (Visible-p (Visible ?object nil)))
 :Main-Acts        ((Look S H)
                    (Make-recognize S H (Object ?object) MM)
                    (Request S H (Try H (action ?act)) NO-SYNC MM))
 :Subsidiary-Acts ((Describe-act S H ?act MM)
                    (Reset S (action ?act))))
```

Figure 4.4: Examples of content plan operators

4.5 Selecting the Content of Instruction Dialogue

In this section, we introduce the CP and show how the instruction dialogue is decided in an incremental way to match the user's view.

4.5.1 Content Planner

In MID-3D, the CP is called by the DM. When a goal is put to the CP from the DM, it selects a plan operator for achieving the goal, applies the operator to find new subgoals, and returns them to the DM. The subgoals are then added to the goal agenda maintained by the DM. Therefore, the CP provides the scenario for the instruction dialogue to the DM and enables MID-3D to output coherent instructions. Moreover, the Content Planer employs depth-first search with a refinement-driven hierarchical linear planning algorithm as in the last chapter as well as in (Cawsey, 1992). The advantage of this method is that the plan is developed incrementally, and can be changed while the conversation is in progress. Thus, by applying this algorithm to 3D dialogues, it becomes possible to set instruction dialogue strategies that are contingent on the user's view.

4.5.2 Considering the User's View in Content Selection

In order to decide the dialogue content according to the user's view, we extend the description of the content plan operator (André and Rist, 1993) by using the

user's view as a constraint in plan operator selection. We also modify the constraint checking functions of the previous planning algorithm such that HyCLASS is queried about the state of the virtual environment.

Figure 4.4 shows examples of content plan operators. Each operator consists of the name of the operator (Header), the effect resulting from plan execution (Effect), the constraints for executing the plan (Constraints), the essential subgoals (Main-acts), and the optional subgoals (Subsidiary-acts). As shown in ⟨Operator 1⟩ in Figure 4.4, we use the constraint `(Visible-p (Visible ?object t))` to check whether the object is visible from the user's viewpoint. Actually, the CP asks HyCLASS to examine whether the object is in the student's field of view.

If an object is bound to the `?object` variable by referring to the knowledge base, and the object is visible to the user, ⟨Operator 1⟩ is selected. As a result, two Main-Acts (looking at the user and requesting to try to do the action) and two Subsidiary-Acts (showing how to do the action, then resetting the state) are set as subgoals and returned to the DM. In contrast, if the object is *not* visible to the user, ⟨Operator 2⟩ is selected. In this case, a goal for making the user identify the object is added to the Main-Acts; `(Make-recognize S H (Object ?object) MM)`.

As shown above, the user's view is considered in deciding the instruction strategy. In addition to the above example, the distance between the target object and the user as well as three dimensional overlapping of objects, can also be considered as constraints related to the user's view.

Although the user's view is also considered in selecting locative expressions of objects in the Sentence Planner in MID-3D, we do not discuss this issue here because surface generation is not the focus of this paper.

4.6 Managing Interruptive Subdialogue

The DM controls the other components of MID-3D based on a discourse model that represents the state of the dialogue. This section describes the DM and shows how the user's view is used in managing the instruction dialogue.

4.6.1 Maintaining the Discourse Model

The DM maintains a discourse model for tracking the state of the dialogue. The discourse model consists of the discourse goal agenda (agenda), focus stack, and dialogue history. The agenda is a list of goals that should be achieved through a dialogue between the user and the system. If all the goals in the agenda are accomplished, the instruction dialogue finishes successfully. The focus stack is a stack of discourse segment frames (DSF). Each DSF is a frame structure that stores the following information as slot values:

– *utterance content (UC):* Semantic representation of utterances constructing a discourse segment. Physical actions are also regarded as utterance contents (Ferguson and Allen, 1998).
– *discourse purpose (DP):* The purpose of a discourse segment.
– *goal state (GS):* A state (or states) which should be accomplished to achieve the discourse purpose of the segment.

In addition to these, we add the user's viewpoint slot to the DSF description in order to track the user's viewpoint information:
– *user's viewpoint (UV):* Current user's viewpoint, which is represented as the position and orientation of the camera. The position consists of x-, y-, and z-coordinates. The orientation consists of x-, y-, and z-angles of the camera.

The basic algorithm of the DM is to repeat (a) the performing actions step and (b) updating the discourse model, until there is no unsatisfied goal in the agenda (Traum, 1994). In performing actions step, the DM decides what to do next in the current dialogue state, and then performs the action. When continuing the system explanation, the DM posts the first goal in the agenda to the CP. If the user's response is needed in the current state, the DM waits for the user's input.

The other step in the DM algorithm is to update the discourse model according to the state that results from the actions performed by the user as well as the actions performed by the system. Although we do not detail this step here, the following operations could be executed depending on the case. If the current discourse purpose is accomplished, the top level DSF is popped and added to the dialogue history. The system then assumes that the user understands the instruction and adds the assumption to the user model. If a new discourse purpose is introduced from the CP, the DM creates a new DSF by setting the header of the selected plan operator in the discourse purpose slot and the effect of the operator in the goal state slot. The DSF is then pushed to the focus stack. If the current discourse purpose is continued, the DM updates the information of the top level DSF.

4.6.2 Considering the User's View in Coping with Interruptive Subdialogues

The main difference of the Dialogue Manager of our system from the previous one is to maintain the user's viewpoint information and use this in managing the dialogue. When the DM updates the information of the current DSF, it observes the user's viewpoint at that point and renews the UV slot and it also adds the semantic representation of utterance (or action) in the UC slot. As a result, it becomes possible to update the user's viewpoint information at each turn, and to track the user's viewpoint in an on-going dialogue.

By using this mechanism, the DM can cope with interruptive subdialogues. In

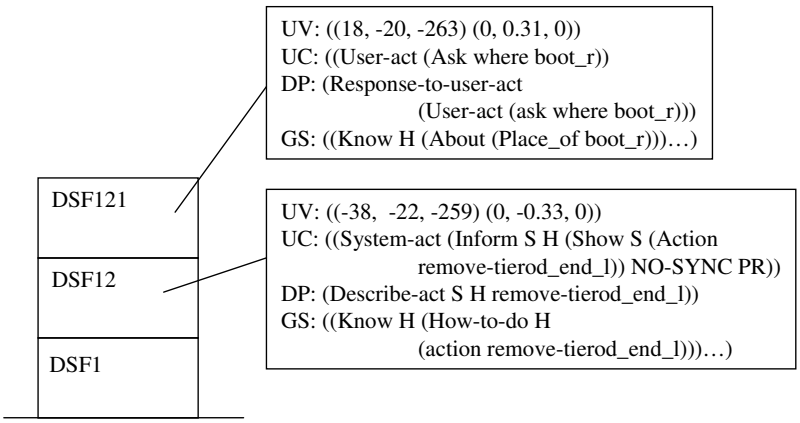


Figure 4.5: Example of the state of a dialogue

resuming from a subdialogue, the user may become confused if the dialogue is resumed but the observed state differs from what the user remembers. In order to match the view to the resumed dialogue, the DM refers the UV slot of the top DSF and puts the users view back to that point. This ensures that the user experiences a smooth transition back to the previous topic. Figure 4.5 shows an example of the state of a dialogue. DSF12 represents a discourse segment that describes how to remove the left tie rod end. DSF121 represents the user-initiated interruptive subdialogue about where the right boot is. Immediately before starting DSF121, the user's viewpoint in DSF12 is ((-38, -22, -259) (0, -0.33, 0)). After completing the subdialogue by answering the user's question, DSF121 is popped and the system resumes DSF12. At this time, the DM gets the viewpoint value of the top DSF (DSF12), and commands HyCLASS to change the viewpoint to that view, which is in this case ((-38, -22, -259) (0, -0.33, 0)) [1] . The system then restarts the interrupted dialogue.

4.7 Example

In order to illustrate the behavior of MID-3D, an example is shown in Figure 4.6. This is a part of an instruction dialogue on how to dismantle the steering system of a car. The current topic is removing the left knuckle arm. In utterance [14], the system describes how to remove this part in conjunction with an animation created by HyCLASS.

In utterance [15], the user interrupted the system's instruction and asked "What is this?" by clicking the right knuckle arm. At this point, the user's speech input

[1] In the current system, it is not possible to move the camera to an arbitrary point because of the limitations of the virtual environment controller employed. Accordingly, this function is approximated by selecting the nearest of several predefined viewpoints.

[14]System: The left knuckle arm is removed like this.
(with the animation showing the left knuckle arm coming off)
[15]User: (After moving the viewpoint to Figure1 and clicking the right knuckle arm)
What is this?
[16]System: This is the right knuckle arm.
[17]User: OK.
[18]System: Now, let's continue the explanation.
(with changing the view to the one in utterance [14])
[19]System: The left knuckle arm is removed like this.
(with the animation showing the left knuckle arm coming off)
[20]User: Yes.
[21]System: Now, try removing the left knuckle arm.
[22]User: (After clicking the left knuckle arm) Remove it.
[23]System: You have successfully completed the operation.
[24]System: The next step is to remove the right tie rod end.
[25]System: Let's move to the right tie rod end.
[26] User: Yes.
[27]System: (Moving the view to the right)
The right tie rod end is in front of you.
[28] User: Yes, I see.

Figure 4.6: Example of a dialogue with MID-3D

was interpreted in the Input Analyzer and a user initiative subdialogue started by pushing another DSF onto the focus stack. In order to answer the question, the DM asked the Domain Plan Reasoner how to answer the user's question. As a result, a discourse goal was returned to the DM and added to the agenda. The DM then sent the goal `(Describe-name S H (object knuckle_arm_r))` to the CP. This goal generated utterance [16].

In system utterance [18], in order to resume the dialogue, a meta-comment, "Now let's continue the explanation", was generated and the viewpoint returned to the previous one in [14] as noted in the DSF. After returning to the previous view, the interrupted goal was re-planned. As a result, utterance [19] was generated.

After completing this operation in [23], the next step, removing the right tie rod end, is started. At this time, if the user is viewing the left side (Figure 4.2) and the system has the goal `(Instruct-act S H remove-tierod_end_r MM)`, ⟨Operator 2⟩ in Figure 4.4 is applied because the target object, right tie rod end, is not visible from the user's viewpoint. Thus, a goal of making the user view the right tie rod end is added as a subgoal and utterances [24] and [25] are generated.

4.8 Summary and Discussion

This chapter proposed a method for altering instruction dialogues to match the user's view in a virtual environment. We described the Content Planner which can incrementally decide coherent instruction dialogue content to match changes in the user's view. We also presented the Dialogue Manager, which can keep track of the user's viewpoint in an on-going dialogue and use this information in resuming from interruptive subdialogues. These mechanisms allow to detect mismatches between the user's viewpoint and the topic at any point in the dialogue, and then to choose the instruction content and user's viewpoint appropriately. MID-3D, an experimental system that uses these mechanisms, shows that the method we proposed is effective in realizing instruction dialogues that suit the user's view in virtual environments.

Chapter 5

Dialogue Management Using Nonverbal Signals

As the other primary component of the *Conceptualization Module* in the MCI architecture, this chapter focuses on the *Dialogue State Manager*, which consists of the *Grounding process* and the *Dialogue State Updating Process*.

First, in an empirical study of the grounding process in human face-to-face conversation, we report on the results of an investigation into the relationship between verbal and nonverbal means for establishing common ground. Previous studies have revealed various functions of nonverbal signals. However, the role of nonverbal signals in grounding has been little investigated. We analyzed eye gaze, head nods and attentional focus in the context of a direction-giving task in which the face of the interlocutors was or was not visible. We found a surprising overall pattern of monitoring lack of negative feedback, and also that the distribution of nonverbal behaviors differs depending on the type of speech act being grounded.

Based on these results, we propose a design for embodied conversational agents (ECAs) that relies on both verbal and nonverbal signals to establish common ground in human-computer interaction. Then, we present an embodied conversational agent that can recognize and generate verbal and nonverbal grounding acts, and use them in updating the discourse state. Finally, a preliminary evaluation shows that the usage of nonverbal behaviors in interaction between our agent and a user is strikingly similar to that in our empirical study of human-human communication. The results strongly support our model, and demonstrate a possible style of future human-computer interaction.

In the following sections, first, we describe the problems addressed in this chapter, then discuss relevant previous work. In Section 5.3, we report results from our own empirical study and, based on our analysis of conversational data. Section 5.4 proposes a model of face-to-face grounding using both verbal and nonverbal

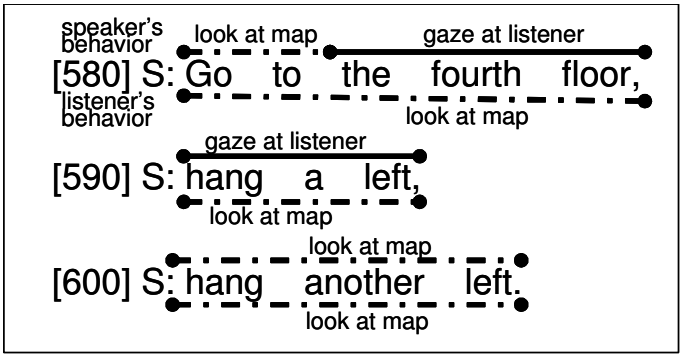


Figure 5.1: Human face-to-face conversation

information. Section 5.5 presents our implementation of that model into an ECA. As a preliminary evaluation, in Section 5.6, we compare a user interacting with the ECA with and without grounding.

5.1 Problem

An essential part of conversation is to ensure that the other participants share an understanding of what has been said, and what is meant. The process of ensuring that understanding - adding what has been said to the common ground - is called grounding (Clark and Schaefer, 1989). In face-to-face interaction, nonverbal signals as well as verbal participate in the grounding process, to indicate that an utterance is grounded, or that further work is needed to ground.

Figure 5.1 shows an example of human face-to-face conversation. Even though no verbal feedback is provided, the speaker (S) continues to add to the directions. Intriguingly, the listener gives no explicit nonverbal feedback - no nods or gaze towards S. S, however, is clearly monitoring the listener's behavior, as we see by the fact that S looks at her twice (continuous lines above the words). In fact, our analyses show that maintaining focus of attention on the task (dash-dot lines underneath the words) is the listener's public signal of understanding S's utterance sufficiently for the task at hand. Because S is manifestly attending to this signal, the signal allows the two jointly to recognize S's contribution as grounded.

Because of its significance as a basis of human communication, grounding has received significant attention in the literature. However, previous work has not addressed some essential questions. First, no previous study has proposed a model of face-to-face grounding that accounts for how people use nonverbal signals combined with verbal acts to ground information. Nonverbal signals do not convey meaning,

propositional content, by themselves, but seem to be integrated into verbal behaviors to serve the purpose of coordinating the interaction. This type of function of nonverbal behaviors is referred as interactional function (Bavelas et al., 1995; Cassell et al., 2000). Thus, interpreting interactional function of nonverbal behaviors on the basis of verbal communication would be a key of establishing a model of grounding that accounts for how verbal and nonverbal behaviors interact with each other to ground information.

Second, according to the discussion above, in human communication there is no doubt that nonverbal signals play important roles in a grounding process. Do they also contribute to improving naturalness of human-computer interaction? If people can ground information with a computer system in the same way as they do with other person, such human interface will improve naturalness of human-computer interaction, and be expected to reduce the burden of using a computer system (Reeves and Nass, 1996). As a human interface which leverages human communication protocols in human-computer interaction, we have proposed and been developing Embodied Conversational Agents (ECAs) (Cassell, 2000), which are animated computer agents capable of multimodal face-to-face conversation with users, including hand gesture, gaze, intonation, and body posture. As an extension of our previous systems, we propose a design of an ECA who can recognize and generate nonverbal signals for grounding based on a model of human face-to-face grounding.

In summary, this chapter addresses the following issues with the goal of contributing to the literature on discourse phenomena, and of building more advanced conversational humanoids that can engage in human conversational protocols:

1. what is a model of face-to-face grounding that accounts for how verbal and nonverbal behaviors interact with each other in grounding?

2. if a model of face-to-face grounding is successfully established, then how can the model be used to adapt dialogue management to face-to-face conversation with an embodied conversational agent?

The outcome of this study is to provide empirical support for an essential role of nonverbal behaviors in grounding, motivating an architecture for an embodied conversational agent that can establish common ground using eye gaze, head nods, and attentional focus.

5.2 Previous work

5.2.1 Models of Grounding

Conversation can be seen as a collaborative activity to accomplish information-sharing and to pursue joint goals and tasks. Under this view, agreeing on what has been said, and what is meant, is crucial to conversation. The part of what has been said that the interlocutors understand to be mutually shared is called the common ground, and the process of establishing parts of the conversation as shared is called grounding (Clark and Schaefer, 1989; Clark and Wilkes-Gibbs, 1986). They defined contribution as a unit of grounding. In their model, contribution is composed of two main phases: in presentation phase, speaker A presents utterance u, and in acceptance phase, interlocutor B accepts u by giving evidence to A, that he believes he understands what A means by u. According to (Clark, 1996; Clark and Schaefer, 1989), eye gaze is the most basic form of positive evidence that the addressee is attending to the speaker. Head nods have a similar function to verbal acknowledgements such as "uh huh", "I see". They suggest that nonverbal behaviors mainly contribute to lower levels of grounding, to signify that interlocutors have access to each other's communicative actions, and are attending.

As a part of the theory of grounding, Clark and Wilkes-Gibbs (1986) also proposed a Principle of Least Collaborative Effort: conversational participants attempt to minimize the effort expended in grounding. Conversants do not always convey all the information at their disposal. Sometimes it takes less effort to produce an incomplete contributions that can be repaired if needs be. Grice (1975) expressed this idea in terms of two maxims; Quantity (make your contribution as informative as is required for the current purpose of the exchange, but do not make your contribution more informative than is required), and Manner (Be brief, and avoid unnecessary prolixity). Since this principle suggests that the way of displaying evidence of understanding, which would reflects communication effort, may be different depending on the communication medium. Clark and Brennan (1991) actually claimed that the way of displaying positive evidence of understanding is different depending on communication modality. In face-to-face conversation, it is easy to nod at interlocutors, and to gaze at interlocutors to show them that they are being attended to, or to monitor their facial expressions. In media without co-presence, nonverbal signals cost expensive bandwidth, or are severely limited.

Modifying the Clark's original theory, Traum (1994) has proposed a computational approach to grounding where the status of contributions as provisional or shared is part of the dialogue system's representation of the "information state" of the conversation (Matheson, Poesio, and Traum, 2000). In his Grounding Acts Model, rather than the two phases of presentation and acceptance, the basic build-

ing blocks are a set of Grounding Acts, each of which is identified with a particular utterance unit, and performs a specific function towards the achievement of common ground. Moreover, instead of contributions, the units of grounded content are Discourse Units (DU). Individual grounding acts triggers updates that register provisional information as a shared DU, and achieve grounding. Acknowledgment acts are directly associated with grounding updates while other utterances effect grounding updates indirectly, because they proceed with the task in a way that presupposes that prior utterances are uncontroversial. Based on this claim, Traum (1994) proposed a DU state transition diagram, which defines possible sequence of grounding acts to achieve common ground (see Section 2.4.3).

Paek and Horvitz (1999), on the other hand, suggest that actions in conversation give probabilistic evidence of understanding, which is represented on a par with other uncertainties in the dialogue system. For example, a speech recognizer is not so accurate and reliable in a certain condition. They provided a dialogue manager that recognizes failures in dialogue, as well as representations and control strategies for grounding using Bayesian networks and decision theory. The dialogue manager assumes that content is grounded as long as it judges the risk of misunderstanding as acceptable.

5.2.2 Nonverbal information as evidence of understanding

A number of studies of face-to-face communication have mentioned that listeners return feedbacks as to whether conversation is on the right track, by giving visual evidence in the form of head nods and attention (Argyle and Cook, 1976; Clark, 1996; Clark and Schaefer, 1989; Duncan, 1972; Duncan, 1974; Kendon, 1967; Rosenfeld and Hancks, 1980). Based on his detailed study of gaze, Kendon (1967) claimed that speakers look up at grammatical pauses to obtain feedback on how utterances are being received, and to see if listeners are willing to carry on the conversation. Goodwin (1981) claimed speakers will pause and restart until they obtain the listener's gaze. By contrast, Novick, Hansen, and Ward (1996) reported that during conversational difficulties, mutual gaze was held longer at turn boundaries. These results suggest that speakers distinguish different patterns of listener's gaze. In some cases, gaze and mutual gaze may be perceived as positive evidence of understanding. In other cases, they may be negative evidence. In fact, Argyle et al. (1973) reported that gaze is used in order to send positive feedback accompanied by nods, and smiles etc, as well as collect information from the partner.

There are some other results evoking another argument for visual evidence. Argyle and Graham (1977) reported that, comparing a situation without any shared object between the conversational participants and one that complex objects (e.g. a map) are shared during the conversation, the percentage of gaze at the other drops

from 76.6% to 6.4%. Similarly, Anderson et al. (1997) reported that mutual gaze falls to below 5% in such a situation. On the basis of these results, (Whittaker, 2003; Whittaker and O'Conaill, 1993) claimed that sharing the same physical environment is important when tasks require complex reference to, and joint manipulations of, physical objects. In a shared environment, speakers and listeners can achieve joint attention to an object or event. If both participants have observed a change to an object or event, they can assume that such changes are part of the conversational common ground. They therefore do not have to be mentioned explicitly.

As for head nods, Duncan (1974) claimed that head nod is a visual back-channel signal provided by listeners in order to provide speakers with useful information while a speaker's turn progresses. Therefore, a head nod does not constitute a speaking turn or a claim of the turn. As more precise investigation of head nod, Rosenfeld and Hancks (1980) attempted to subcategorize functions of listener's feedback or back-channel behaviors. They found that listeners expressed "agreement" with complex verbal responses and multiple head nods, while expressed "understanding" with repeated small head nods prior to the speech juncture.

5.2.3 Visual Information in Mediated Communication

Studies in mediated communication also made a great deal of contribution to theories of face-to-face communication. They isolated effects of different behaviors (e.g. facial expressions, gaze, gesture) by employing experimental research methods, and provided comparable data for clarifying the role of each behavior (Whittaker, 2003). Brennan (2000) provides experimental evidence that shows how communication modality affects the cost for accomplishing the common ground. When more direct evidence, such as listener's displaying correct task manipulation, was available, the grounding process became shorter. As a similar experimental study for multimodal communication, Dillenbourg, Traum, and Schneider (1996) found that grounding is often performed across different modes. The subjects are in a virtual environment where they can use three modes of communication: verbal communication, action command for changing the virtual environment, and whiteboard drawing. The results showed that information presented verbally was grounded by an action in the virtual environment. Also, actions in the virtual environment were grounded verbally.

Research in video mediated communication (VMC), which aims at proposing technologies supporting mediated communication with visual information, has also highlighted the importance of nonverbal signals. In videoconferencing system, visual information usually displays head movements, gaze, and facial expressions of conversational participants, and these behaviors indicate how speaker's utterances have been received. Comparing "map task" conversations in audio-only and VMC where

the subjects can make direct eye contact, Boyle, Anderson, and Newlands (1994) found that speakers more frequently check listeners' understanding verbally when they only have an audio link than when visual signals are available. Similarly, comparing audio and video conference condition, Daly-Jones, Monk, and Watts (1998) reported that interpersonal awareness was much increased in the video mediated communication than the audio condition. These results suggest that availability and quality of a visual channel conveying nonverbal signals affects the grounding process in VMC as well as human face-to-face conversation. Moreover, studies in collaborative virtual environments (CVEs) reported that task performance is much higher when subjects are gazed at by their partners whenever speaking or being listened to, as opposed to randomly (Garau et al., 2001; Vertegaal and Ding, 2002).

Not only nonverbal signals directed to conversational partners, but also those to a shared environment may serve as evidence of understanding. Whittaker and O'Conaill (1993) and Whittaker (2003) pointed out the importance of shared objects serving as implicit common ground in a shared environment. Whittaker and O'Conaill (1993) compared communication effectiveness with and without a shared workspace. The shared workspace enabled people to share visual material such as documents or designs as well as to type, draw, and write. They reported that participants with the shared workspace took fewer turns for identifying pieces, because they were able to refer to pieces deictically. Similar results were reported in comparing speech, speech/video, and face-to-face communication (Kraut, Miller, and Siegel, 1996; Olson, Olson, and Meader, 1995).

5.2.4 Nonverbal Behaviors of Animated Agents

The significant advances in computer graphics over the last decade has enabled implementing animated agents capable of face-to-face interaction with human users. The agents display nonverbal behaviors of different functions for different purposes. First, presentation agents (André, Rist, and Muller, 1999; Noma and Badler, 1997; Wahlster, Reithinger, and Blocher, 2001) attempt to emulate presentation styles common in human-human communication (André, 2000). For example, these agents are able to point towards objects on the screen using deictic gestures or a pointer so as to demonstrate comprehensible explanation about multimedia contents.

The second area, pedagogical agents cohabit learning environment with students to create rich, face-to-face learning interactions (Johnson, Rickel, and Lester, 2000a). In pedagogical agents, nonverbal behaviors are primarily used to enhance the quality of advice to learners. The agents demonstrate the task to users by manipulating objects in a virtual environment (Rickel and Johnson, 1999), and navigate the user's attention using pointing and gaze (Lester et al., 1999; Rickel and Johnson, 1999). They also nonverbally express the evaluation of learner's performance, using nod,

shake, and puzzled, pleasant, or more exaggerated facial expressions (Lester et al., 1999; Rickel and Johnson, 1999; Shaw, Johnson, and Ganeshan, 1999; Stone and Lester, 1996). More recently, Traum and Rickel (2002; Traum et al. (2003) proposed a multiparty conversational environment where multiple animated agents talk with each other on the screen and one of them can communicate with a user.

The third area, which is close to this study, is Embodied Conversational Agent (ECA). Research in ECA is attempting to improve naturalness of human-computer interaction by implementing face-to-face conversational protocols in animated agents (Cassell et al., 2001). Agents mimic human communicative nonverbal behaviors using their face and body. For example, agents display eyebrow raise and beat gestures to emphasize intonationally salient words in speech (Hadar, 1989), and change gaze direction for floor management (Duncan, 1974). In their ECA evaluation experiment, (Cassell and Thorisson, 1999) demonstrated that correct relationships among verbal and nonverbal signals enhances the naturalness and effectiveness of embodied dialogue systems. They reported that users felt the agent to be more helpful, lifelike, and smooth in its interaction style when it demonstrated nonverbal conversational behaviors.

5.2.5 Our Approach

Previous theories of grounding describe that head nod and attention are basic forms of positive evidence of grounding. By contrast, studies of human face-to-face or mediated communication suggest that functions of nonverbal behaviors may be different depending on the physical as well as linguistic context of conversation. Mutual gaze may indicate conversational difficulties. Attention to a shared environment may serve as positive evidence specifically when people are engaging in a joint task. Therefore, this study focuses on a physical conversational situation where conversational participants are engaged in a join task, and investigate how nonverbal behaviors interact with verbal behaviors there. Then, we will establish empirical model of face-to-face grounding based on the data analysis, and apply the model to an ECA.

By following the approach described here, this chapter will provide clear answers for the issues: (1) how to establish a model of face-to-face grounding with respect to interaction between verbal and nonverbal behaviors, (2) how to apply the model of face-to-face grounding to human-computer interaction.

5.3 Empirical Study

This section describes our empirical study of usage of nonverbal behaviors in face-to-face conversation. The data analysis will provide a basis for modeling face-to-face grounding, and implementing an ECA.

5.3.1 Experiment

Design and subjects

Ten students or employees in the MIT Media Laboratory and ten students outside of the lab, who did not know the floor plans of the Media Lab building, were paired. A subject from the lab (a direction giver) gave a direction to somewhere in the lab to a student outside of the lab (a direction receiver), who did not know about it at all. Each pair of subjects had a conversation in each of the following conditions.

(1) Face-to-face condition (F2F): where two subjects sat with a map drawn by the direction-giver. The subjects share the map, and also see the other's face and body.

(2) Shared Reference condition (SR): where an L-shaped screen between the subjects let them share a map drawn by the direction-giver, but not to see the other's face or body.

Procedure and Instructions

Drawing a map of a route by a direction giver: Before the session, an experimenter asked the giver to draw two maps of what s/he would explain using at least 8 landmarks or signs. The instruction given to the givers is shown in the Appendix.

Direction giving task: Then, each pair of subjects is engaged in two conversations in two different experimental settings. Instruction for F2F condition is shown in the Appendix.

Data storage

Interactions between the subjects were shot from four different angles, the pictures were combined by a video mixer into synchronized video clips, and video-recorded with a SVHS recorder. A snapshot of an experiment session in F2F condition is shown in Figure 5.2. Camera (A) shows a shared map and the movement of subjects' fingers, Camera (B) shows a close up picture of a receiver, Camera (C) shows a close up picture of a giver, and Camera (D) shows an overall picture of the interaction.

5.3.2 Data Coding

By running 10 experiment sessions, 10 dialogues per condition (20 in total) were collected and transcribed.

Figure 5.2: Snapshot of experiment session

Coding verbal behaviors

As grounding occurs within a turn, which consists of consecutive utterances by a speaker, following Nakatani and Traum (1999), we tokenized a turn into utterance units (UU), corresponding to a single intonational phrase (Pierrehumbert, 1980). Each UU was categorized using the DAMSL coding scheme (Allen and Core, 1997). The advantage of using this coding scheme is that the inter-coder reliability of the scheme has already been reported (Core and Allen, 1997). Their report provides good support that the quality of our data is good enough though we did not calculate the inter-code reliability for our data.

In the statistical analysis, we concentrated on the following four categories with regular occurrence in our data:

Acknowledgement: verbal actions consisting of short phrases such as "okay", "yes", and "uh-huh", and signaling that the previous utterance was understood.

Information request (Info-req): verbal actions that introduces an obligation to provide an answer

Answer: verbal actions serving as an aspect of a binary dimension where UUs can be marked as complying with an information request action in the antecedent

Assertion: statements which are uttered in trying to affect the beliefs of the hearer

Table 5.1: NV statuses

Combinations of NVs		Listener's behavior			
		gP	gM	gMwN	gE
Speaker's behavior	gP	gP/gP	gP/gM	gP/gMwN	gP/gE
	gM	gM/gP	gM/gM	gM/gMwN	gM/gE
	gMwN	gMwN/gP	gMwN/gM	gMwN/gMwN	gMwN/gE
	gE	gE/gP	gE/gM	gE/gMwN	gE/gE

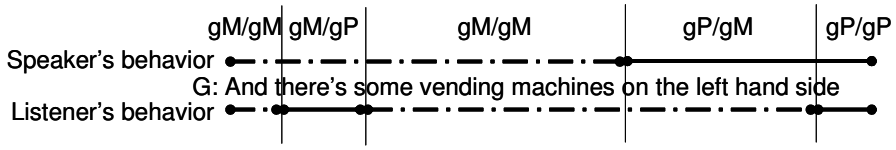


Figure 5.3: Example of coding NV status

Coding nonverbal behaviors

As nonverbal data, we analyzed eye-gaze and head nod. The definition of gaze is based on (Exline and Fehr, 1982) and categories of head movement were extracted from the body movement scoring system proposed by Bull (1987). Gaze at Partner (gP): Looking at the partner's eyes, eye region, or face.

Gaze at Map (gM): Looking at the shared map

Gaze Elsewhere (gE): Looking away elsewhere

Head nod (Nod): Head moves up and down in a single continuous movement on a vertical axis, but eyes do not go above the horizontal axis.

By combining Gaze and Nod, six complex categories, such as gP with nod, and gP without nod, are generated. For example, gaze at the map with nod is coded as gMwN. In what follows, however, we analyze only categories with more than 10 instances. In addition, in order to analyze dyadic behaviors of conversational participants, 16 combinations of the nonverbal behaviors by a speaker and a listener are defined, as shown in Table 5.1. For instance, gP/gM stands for a combination of speaker gaze at the partner and listener gaze at the map, and gP/gMwN stands for that of speaker gaze at the partner and listener gaze at the map while nodding. Then, these combination categories are used to describe how dyad's nonverbal status changes during the process of grounding. The example in Figure 5.3 shows nonverbal

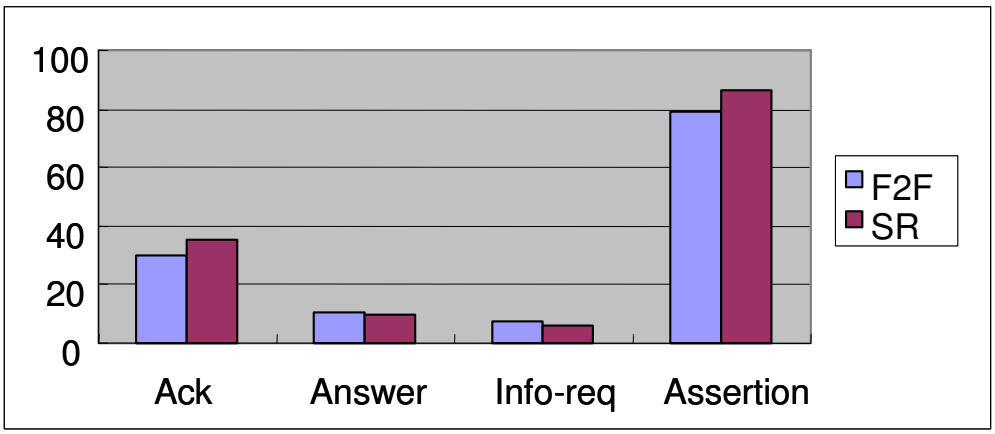


Figure 5.4: Mean number of the four types of UUs per dialogue

behaviors during an utterance by the direction giver (G). Dash-dot lines indicate the place of gaze at the map (gM), and continuous lines indicate those of gaze at the partner (gP). In this example, the NV status at the beginning of the UU is gM/gM. It shifts to gM/gP, then back to gM/gM. Then, the speaker starts gazing at the listener (gP/gM), and the listener also starts gazing at the speaker at the very end of the UU, where they get mutual gaze (gP/gP). The mutual gaze continues after the UU. As quantitative data, we count the number of shifts occurs within and between a UU, and analyze them statistically. For example, NV status transition from gP/gP to gM/gM is counted as one shift[1].

5.3.3 Analysis

Analysis 1: Comparison between face-to-face (F2F) and shared reference (SR) condition

First, we compare general characteristics of communication between F2F and SR by reporting descriptive statistics for verbal and nonverbal behaviors.

Results

The analyzed corpus consists of 1088 UUs for F2F, and 1145 UUs for SR. The mean length of conversations in F2F is 3.24 minutes, and in SR is 3.78 minutes ($t(7) = -1.667$ $p < .07$ (one-tail)). The mean length of utterances in F2F (5.26 words per UU) is significantly longer than in SR (4.43 words per UU) ($t(7) = 3.389$

[1]To look at the continuity of NV status, we also analyzed the amount of time spent in each NV status. For gaze, transition and time spent gave similar results. Since head nods are so brief, however, we discuss the data in terms of transitions.

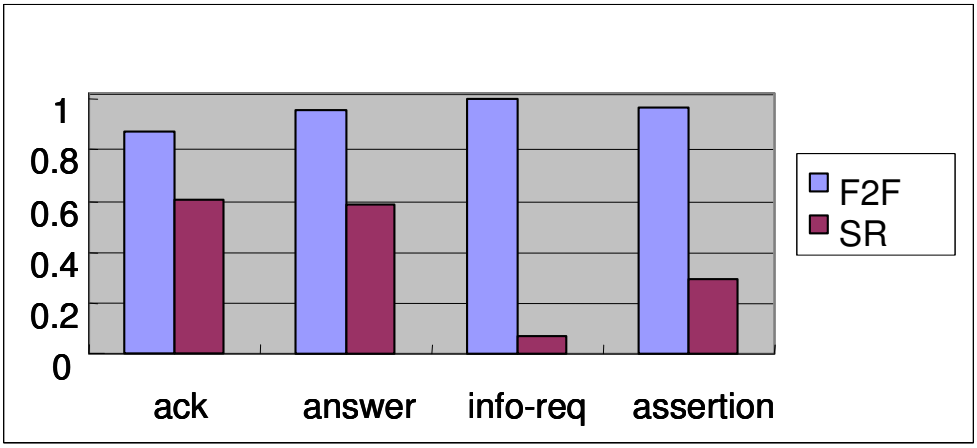


Figure 5.5: Mean number of NV status shifts occurring in each type of UU

$p < .01$ (one-tail)). The mean number of the four types of UUs per dialogue is shown in Figure 5.4. In all types of UUs, the frequency is not statistically different depending on the communication modality. As for the nonverbal behaviors, the number of shifts between the statuses in Table 5.1 was compared. The total number of NV status shifts for F2F is 887 and 425 for SR ($t(7) = 3.377$ $p < .01$ (one-tail)). The number of NV status shifts in SR is less than half of that in F2F, and the difference is statistically significant. Figure 5.5 shows the mean number of NV status shifts occurring in each type of UU. This shows a probability of occurrence of an NV status shift per UU. For all types of verbal act, the probability of NV status shift occurrence is significantly higher in F2F than in SR (Acknowledgement: $z = 6.814$ $p < .01$, Answer: $z = 5.698$ $p < .01$, Info-req: $z = 9.614$ $p < .01$, Assertion: $z = 25.19$ $p < .01$ [2]).

Discussion

Boyle, Anderson, and Newlands (1994) compared map task dialogues between two conditions: the conversational participants can see each other's face and they cannot see each other's face. They found that conversational participants who could not see each other produced more turns (longer dialogues) than those who could see each other. Although we did not count the number of turns, we got a similar result with respect to time: the time length of conversation in SR is longer than in F2F. They also reported that speakers who could not see their partners used fewer word tokens per turn than those who could see each other. We got the same result concerning this point; the speakers produce fewer words per UU in SR than F2F condition. These results suggest that, in SR, where the mean length of UU is shorter, speakers

[2]For testing the difference of means, z score is used.

Table 5.2: Salient transitions

	Shift to	
	within UU	pause
Acknowledgement	*gMwN/gM* (0.495)	*gM/gM* (0.888)
Answer	*gP/gP* (0.436)	*gM/gM* (0.667)
Info-req	*gP/gM* (0.38)	*gP/gP* (0.5)
Assertion	*gP/gM* (0.317)	*gM/gM* (0.418)

present information in smaller chunks than in F2F, leading to more chunks and a slightly longer conversation. In F2F, on the other hand, conversational participants convey more information in each UU.

As for nonverbal behaviors, the total number of NV status shifts per dialogue in SR is less than half of that in F2F. Although (Boyle, Anderson, and Newlands, 1994) did not provide comparison of nonverbal behaviors between conditions, our result was clear enough, suggesting that nonverbal behaviors in F2F are used as signals serving interactive function in a communication with co-presence. Thus, all these results of comparison between F2F and SR indicate that visual access to the interlocutor's body affects the conversation in terms of usage of nonverbal as well as verbal acts.

Analysis 2: Correlation between verbal and nonverbal behaviors

In Analysis 1, we confirmed that nonverbal behaviors, such as eye-gaze and head nod, serve interactive function in F2F communication. In Analysis 2, we will explore interaction between verbal and nonverbal behaviors, and examine whether nonverbal behaviors signaling positive evidence of understanding are different depending on the type of verbal act. If we can find a clear relationship between speakers' verbal act and listeners' nonverbal behaviors, speakers' verbal act can be used for predicting nonverbal positive evidence displayed by listeners. In addition, we will also investigate whether, for each type of verbal act, the usage of nonverbal behaviors is different depending on the experimental condition (F2F/SR), and whether the effect of communication modality is different depending on the type of verbal act. We will address these issues by analyzing NV status shifts with respect to the type of verbal communicative action.

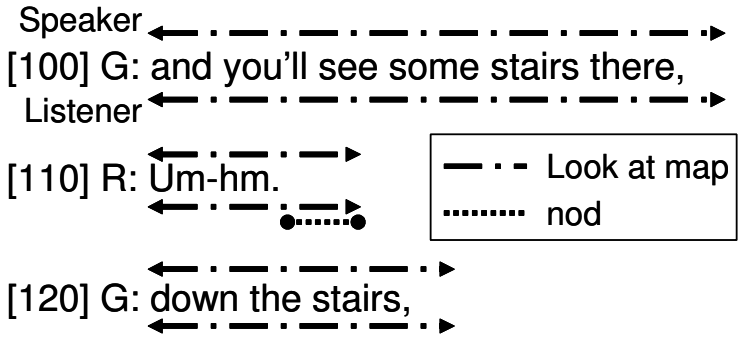


Figure 5.6: Example of non-verbal acts in Acknowledgement

Results

Table 5.2 shows the most frequent target NV status (shift to these statuses from others) for each speech act type in F2F. Numbers in parentheses indicates the proportion to the total number of transitions. The table shows that the most frequent NV status transition pattern is different depending on the type of verbal act. As the subjects rarely demonstrated communication failures during the conversation, we can assume that the most frequent NV status represent nonverbal signals exchanged between conversational participants when information is successfully grounded. Likewise, listeners' most frequent nonverbal behaviors represent positive evidence of understanding. More details are described below.

<**Acknowledgement**> An example of a typical interaction is shown in Figure 5.6. "G" indicates that the speaker is a direction giver. "R" indicates that the speaker is a direction receiver. Lines on the upper side of the words show G's nonverbal acts. Lines drawn at the bottom of the words shows R's. At UU [100], both of the conversational participants look at the map (dash-dot lines), and at UU [110], a speaker (receiver) was nodding (dotted line) during acknowledging with "Um-hm", while the listener (giver) looks at the map. Thus, the dyad NV status is gMwN/gM. Then, the speaker stopped nodding after the Acknowledgement, and the dyad NV status shifts to gM/gM. As shown in Table 5.2, this is the typical usage of nonverbal signals accompanied by Acknowledgement. Within an UU, the dyad's NV status most frequently shifts to gMwN/gM (eg., the speaker utters "OK" while nodding, and the listener looks at the map). At pauses, a shift to gM/gM is the most frequent. The same results were found in SR where the listener could not see the speaker's nod.

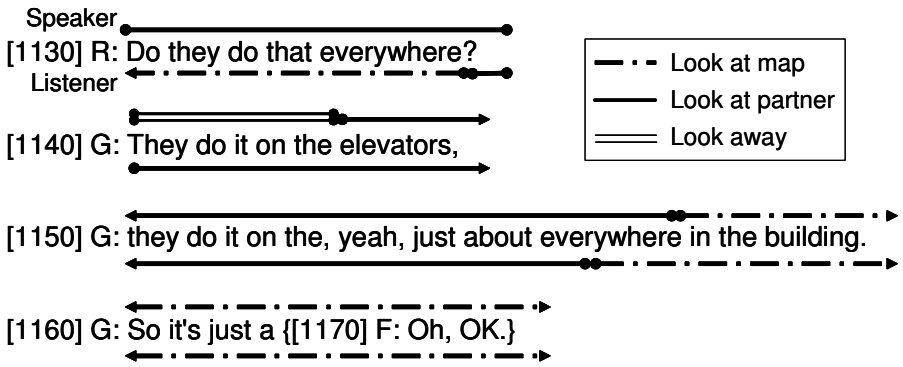


Figure 5.7: Example of non-verbal acts for Info-req and Answer

<**Answer**> An example of a typical interaction is shown in Figure 5.7. Utterance unit [1140], following the direction receiver's Info-req in [1130], is the Answer by the giver. The speaker (the direction giver) looks away (double line) at the beginning of the Answer, and then gazes at the partner. On the other hand, the listener (the direction receiver) keeps looking at the speaker for the whole UU. Then, the giver continues the Answer on to [1150]. At this time, the UU starts with mutual gaze (gP/gP), and then the NV status shifts to gM/gM. As shown in Table 5.2, these are the most frequent shifts within a UU and at pause after the UU respectively.

The results suggest that speakers and listeners rely on mutual gaze (gP/gP) to ensure an answer is grounded. In other words, in answering a question, the speakers appear to need the listener to give their gaze as positive evidence of understanding. However, they cannot use this strategy in SR.

In addition, at UU [1140] in Figure 5.7, the speaker looks away at the beginning of the answer. This finding supports a previous study of human communication by (Argyle and Cook, 1976). They reported that aversion of gaze occurs at the beginning of utterances when cognitively difficult topics are discussed. Therefore, the looking away at the beginning of Answer works as a deliberate signal that the speaker is thinking and planning their reply, which would be perceived by the listener's gaze, and be a sort of display that the current speaker understood and accepted the listener's question.

<**Info-req**> A typical example of Info-req is also shown in Figure 5.7. UU [1130] is an Info-req from the receiver. The NV status starts with gP/gM (the speaker looks at the listener and the listener looks at the map). Then, at a pause after the Info-req, the listener gazes at the speaker, and the dyad status shifts to gP/gP (mutual gaze). As shown in Table 5.2, the most frequent shift within Info-req is to gP/gM, while at pauses between UUs shift to gP/gP is the most frequent. This

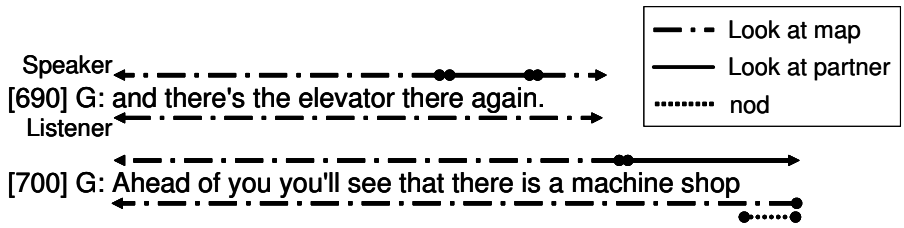


Figure 5.8: Example of non-verbal acts for Assertion

suggests that speakers obtain listeners' gaze after asking a question to ensure that the question is successfully received by the listener, before the turn is transferred to the listener to reply. In SR, however, rarely is there any NV status shift, and participants continue looking at the map.

<**Assertion**> A typical example of Assertion is shown in Figure 5.8. At [690], the speaker (the giver) glances at the receiver while speaking, so that the NV status of the dyad shifts to gP/gM. Then, at a pause after the UU, the speaker's gaze direction moves back to the map, and the NV status of the dyad shifts to gM/gM. These are, again, the typical NV status shifts in Assertion, explored by the statistical analysis.

As for the comparison between F2F and SR, in both conditions listeners look at the map most of the time, and sometimes nod. However, speakers' nonverbal behavior is very different across conditions. In SR, speakers either look at the map or elsewhere. By contrast, in F2F, they frequently look at the listener, as shown in Table 5.2, and a shift to gP/gM is the most frequent within an UU. This suggests that, in F2F, speakers check whether the listener is paying attention to the referent, implying that not only listener's gazing at the speaker, but also paying attention to a referent works as positive evidence of understanding in F2F.

Discussion

First, we got a clear finding that, in F2F condition, the usage of nonverbal behavior is different depending on the type of verbal act. In Answer, the listener's continuous gaze at the speaker during the speaker's answering is required as positive evidence of understanding. In Information request, speakers require to get listener's gaze right after the question. In Assertion, the listener's paying attention to the shared referent (map) serves as evidence of understanding the information conveyed by the speaker's Assertion, suggesting that speakers do not always need the listener's attention and paying attention to the shared map can work as positive evidence by co-occurring with Assertion. Therefore, these findings support our hypothesis that nonverbal behaviors serving as positive evidence of understanding are different

depending on the type of verbal acts. That is, verbal acts can predict nonverbal signals that speakers expect to receive from listeners in a given dialogue context.

On the contrary, in SR condition, NV status shift occurs much less frequently, and we could not find a clear result for the usage of nonverbal behaviors except for Acknowledgement. In Acknowledgement, the typical NV status transition within UU is gMwN/gM in both F2F and SR conditions. When a speaker asserts Acknowledgement, nod almost always accompanies the verbal act. However the listener of the Acknowledgement does not pay attention to this non-verbal signal from the speaker. We also found that the listeners' head nod during the speakers' Assertion is more frequent in F2F than SR ($t(7) = 5.363$ $p < .01$). These results suggest that head nod may function introspectively as well as communicatively. The speaker's nod during her/his own Acknowledgement seems more like an introspective behavior, while the listener's nod during Assertion in F2F is used as a nonverbal display of evidence of understanding directed to speakers.

Analysis 3: Correlation between speaker and listener

Thus far we have demonstrated a difference in distribution among nonverbal behaviors, with respect to conversational action, and visibility of conversation partner. Based on these findings, we found nonverbal signals serving as positive evidence of understanding. However, grounding is not always successful. In order to establish a model that can detect a problem in grounding as well as recognize a success of grounding, it is also necessary to specify nonverbal signals displaying "negative" evidence of understanding.

In Analysis 3, we approach this issue by investigating how listeners' nonverbal behaviors affect speakers' following verbal actions. We look at two consecutive UUs by a direction-giver, and analyze the relationship between the NV status of the first UU and the speaker's (giver's) verbal act in the second UU. The givers' second UUs are classified into the following two categories;

go-ahead: the second UU that introduces a new discourse unit (DU) to be grounded.

elaboration: the second UU that continues the same DU, and gives additional information about the first UU. An example dialogue sequence where S (direction giver) is uttering two consecutive Assertion UUs is shown as follows:

UU1	S: And then, you'll go down this little corridor.
UU2-a	S: It's not very long. (elaboration)
UU2-b	S: Then take a right. (go-ahead)

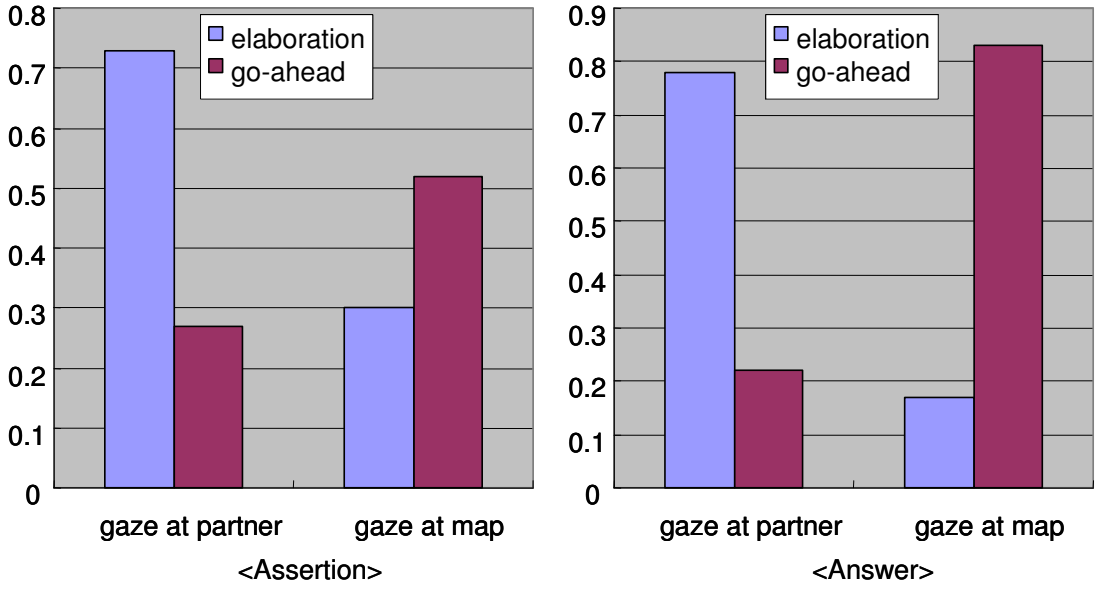


Figure 5.9: Relationship between receiver's NV and giver's next verbal behavior

In the first UU (i.e., [UU1]), the speaker gives a leg of direction with an Assertion speech act. If the second UU is [UU2-a], which gives elaboration about the previous UU, the speaker's second behavior is categorized as "elaboration". If the second UU is [UU2-b], which gives the next leg of the direction, the speaker's second behavior is categorized as "go-ahead".

Results

We did the same analysis for Assertion and Answer because we got enough data for them and these are the most frequent verbal act types generated by our ECA. Results are shown in Figure 5.9. As for Assertion, when the listener begins to gaze at the speaker somewhere within an UU, and maintains gaze until the pause after the UU, the speaker's next UU is an elaboration of the previous UU 73% of the time. On the other hand, when the listener keeps looking at the map during an UU, only 30% of the next UU is an elaboration. The difference in percentage is statistically significant ($z = 3.678$ $p < .01$). Moreover, when the listener keeps looking at the speaker, the speaker's next UU is go-ahead only 27% of the time. By contrast, when the listener keeps looking at the map, the speaker's next UU is go-ahead 52% of the time. The difference is also statistically significant ($z = -2.049$ $p < .05$). The percentage for gaze-at-map does not sum to 100% because some of the UUs are cue phrases or tag questions which are part of the next leg of the direction, but do not convey content.

We also analyzed two consecutive Answer UUs by a giver, and found that when the listener keeps looking at the speaker until a pause, the speaker elaborates the Answer 78% of the time. By contrast, when the listener looks at the speaker during the UU and at the map after the UU, the speaker elaborates only 17% of the time. The difference is statistically significant ($z = -2.324$ $p < .05$).

Discussion

In Assertion, listeners' continuous attention to the map is interpreted as evidence of understanding, and the speakers go ahead to the next leg of the direction. This result supports the findings in the previous section: looking at the map is positive evidence in Assertion. On the other hand, the speakers interpret the listeners' continuous gaze as evidence of not-understanding, and they therefore add more information about the previous UU. Note that similar findings were reported in a study of map task corpus by (Boyle, Anderson, and Newlands, 1994). They reported that during periods of communicative difficulty, direction receivers gazed at the partner more than twice as much as during non-problem points. They discussed that, at times of communicative difficulty, interlocutors are more likely to utilize all the channels available to them.

As for Answer, similar results were found. That is, the listener's continuous gaze at the speaker signals negative evidence of understanding. An interesting example is shown in Figure 5.7. The giver provides two consecutive Answer UUs; [1140] and [1150]. At [1140], the listener (the receiver) keeps gazing at the speaker even at a pause after the UU. This signals evidence of not-understanding. Thus, the speaker gives an elaboration at [1150]. At this time, the listener's gaze shifts to the map, indicating evidence of understanding. However, the direction giver tries to continue on the next UU [1160]. Interestingly, at [1170], the receiver interrupts the giver's UU, and gives verbal evidence of understanding, "Oh, OK. "

5.4 A Model of Face-to-Face Grounding

In this section, based on the empirical findings reported in previous sections, we will establish a new model that accounts for usage of nonverbal behaviors in terms of grounding. Analyzing spoken dialogues, Traum and Heeman (1996) reported that grounding behavior is more likely to occur at an intonational boundary, which we use to identify UUs. This implies that multiple grounding behaviors can occur within a turn if it consists of multiple UUs. However, in previous models, information is grounded only when a listener returns verbal feedback, and acknowledgement marks the smallest scope of grounding. If we apply this model to the example in Figure 5.1, none of the UU has been grounded because the listener has not returned any

spoken grounding clues.

In contrast, our results suggest that considering the role of nonverbal behavior, especially eye-gaze, allows a more fine-grained model of grounding, employing the UU as a unit of grounding. Our results also suggest that speakers are actively monitoring positive evidence of understanding, and also the absence of negative evidence of understanding (that is, signs of miscommunication). When listeners continue to gaze at the task, speakers continue on to the next leg of directions.

Because of the incremental nature of grounding, we implement nonverbal grounding functionality into an embodied conversational agent using a process model that describes steps for a system to judge whether a user understands system contribution:

Step 1 Preparing for the next UU: according to the speech act type of the next UU, nonverbal positive and negative evidence that the agent expects to receive are specified.

Step 2 Monitoring: monitors and checks the user's nonverbal status and signals during the UU. After speaking, the agent continues monitoring until s/he gets enough evidence of understanding or not-understanding represented by user's nonverbal status and signals.

Step 3 Judging: once the agent gets enough evidence, s/he tries to judge groundedness as soon as possible. According to some previous studies, length of pause between UUs is in between 0.4 to 1 sec (Nakajima and Allen, 1992; Traum and Heeman, 1996). Thus, the time out for judgment is 1sec after the end of the UU. If the agent does not have evidence then, the UU remains ungrounded.

This model is based on the information state approach (Matheson, Poesio, and Traum, 2000), with update rules that revise the state of the conversation based on the inputs the system receives. In our case, however, the inputs are sampled continuously, include the nonverbal state, and only some require updates. Other inputs indicate that the last utterance is still pending, and allow the agent to wait further. In particular, task attention over an interval following the utterance triggers grounding. Gaze in the interval means that the contribution stays provisional, and triggers an obligation to elaborate. Likewise, if the system times-out without recognizing any user feedback, the segment remains ungrounded. This process allows the system to keep talking across multiple utterance units without getting any verbal feedback from the user. From the user's perspective, explicit acknowledgement is not necessary, and the minimal cost is involved in eliciting elaboration.

[1] USER: Hello
[2] MACK: Hello, my name is MACK. How are you? You can ask me to describe research groups and projects in the Media Lab. I can also tell you how to get there. So, what would you like to know?
[3] USER: Tell me about Gesture and Narrative Language group.
[4] MACK: Oh, GNL ! They're my favorite group. They build machines that can be a part of face-to-face conversation and story-telling. They're like parents to me!
[5] USER: Oh. Can you tell me about this group?
[6] MACK: Object Based Media conducts research into the future of electronic visual communication.
[7] USER: How would I get to Object Based Media?
[8] MACK: Object Based Media is in the Garden, room 344. We are right here. Go to here, and make a right. Go to this door, and make a left.
[9] USER: OK
[10] MACK: And that's the Garden.
[11] USER: And how do I get to the Jungle?
[12] MACK: We are right here. Go to this door and make a right. It's the door next to the printer and copy machine. Walk to the corner, and make a right. Go to the glass doors, and make a left. And that's the Jungle.
[13] USER: Thanks MACK.
[14] MACK: Pleasure talking to you.

Figure 5.10: Example of conversation with MACK

5.5 Face-to-face Grounding with ECAs

Now we will implement our model of face-to-face grounding, which is based on our empirical results, into an Embodied Conversational Agent (ECA). The basic idea of the system is that a system recognizes user's eye gaze and head nod as nonverbal information, and a dialogue manager of the ECA system can handle the nonverbal information in updating the dialogue state, which describes what is grounded, and what has not been grounded yet.

5.5.1 System Overview

First, we describe a system architecture of our conversational agent, MACK (Media Lab Autonomous Conversational Kiosk). MACK is an interactive public information ECA kiosk. His current knowledge base concerns the activities of the MIT Media Lab; he can answer questions about the lab's research groups, projects, and demos, and give directions to each.

An example of interaction with MACK is shown in Figure 5.10. After greeting each other in [1] and [2], the user asks about a group in [3]. In [4], MACK gives a description of the group. In [5], the user asks about a group by pointing at a room

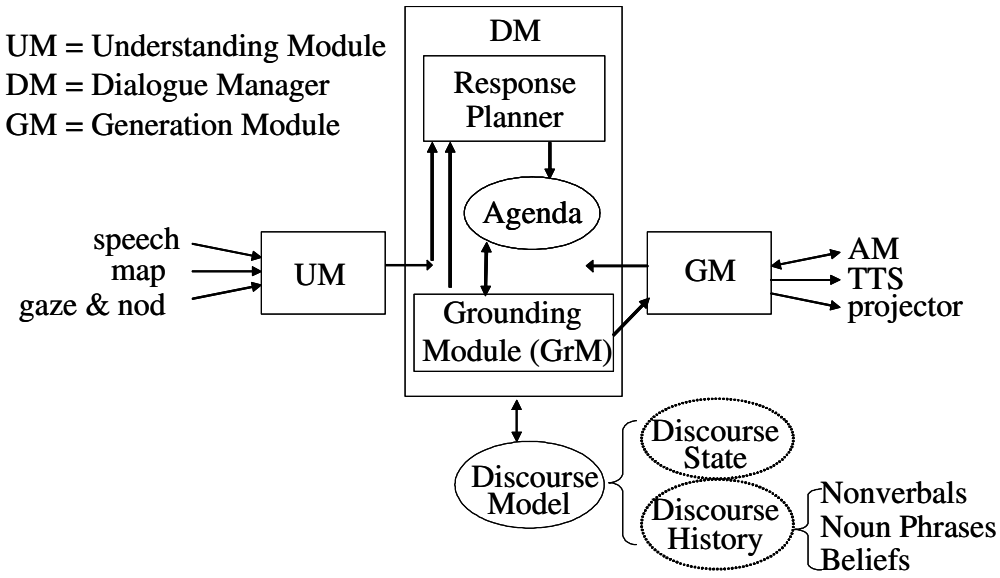


Figure 5.11: MACK system architecture

on a shared map, and MACK answers it in [6]. Then, the user asks how to get to the group in [7]. In [8], after answering the location of the group, MACK gives a direction to the room. In [11], the user asks how to get to a room, "the Jungle", and MACK answers it in [12]. The user and MACK exchange greetings to close the conversation in [13] and [14].

The system architecture is shown in Figure 5.11. The arrows indicate the directions of sending and receiving information between modules. On the input side, MACK recognizes three modalities: (1) speech, using IBM's ViaVoice, (2) pen gesture via a paper map atop a table with an embedded Wacom tablet, and (3) head nod and eye gaze via a stereo-camera-based 6-degree-of-freedom head-pose tracker, based on (Morency, Rahimi, and Darrell, 2003). These inputs operate as parallel threads, allowing the Understanding Module (UM) to interpret the multiple modalities both individually and in combination.

MACK produces multimodal output as well: (1) speech synthesis using the Microsoft Whistler Text-to-Speech (TTS) API, (2) a graphical figure with synchronized hand and arm gestures, and head and eye movements generated by Animation Module (AM), and (3) LCD projector highlighting on the paper map, allowing MACK to reference it.

The UM interprets the input modalities and converts them to dialogue moves which it then passes on to the Dialogue Manager (DM). The DM consists of two primary sub-modules, the Response Planner, which determines MACK's next action(s) and creates a sequence of utterance units, and the Grounding Module (GrM), which

updates the Discourse Model and decides when the Response Planner's next UU should be passed on to the Generation module (GM). The GM converts the UU into speech, gesture, and projector output, sending these synchronized modalities to the TTS engine, Animation Module (AM), and Projector Module.

The Discourse Model maintains information about the state and history of the discourse. This includes a list of grounded beliefs and ungrounded UUs; a history of previous UUs with timestamp; a history of nonverbal information (divided into gaze states and head nods) organized by timestamp; and information about the state of the dialogue, such as the current UU under consideration, and when it started and ended.

5.5.2 Nonverbal Inputs

Eye gaze and head nod inputs are recognized by a head tracker, which calculates rotations and translations in three dimensions based on visual and depth information taken from two cameras (Morency, Rahimi, and Darrell, 2003). The calculated head pose is translated into "look at MACK, " "look at map, " or "look elsewhere. " The rotation of the head is translated into head nods, using a modified version of (Kapoor and Picard, 2001). Head nod and eye gaze events are timestamped and logged within the nonverbal component of the Discourse History. The Grounding Module can thus look up the appropriate nonverbal information to judge a UU.

5.5.3 The Dialogue Manager

The DM decides the agent's next action or UU, and updates the discourse state according to the judgment of grounding. The DM repeats these two processes until the end of the interaction. In a kiosk ECA, the system needs to ensure that the user understands the information provided by the agent. For this reason, we concentrated on implementing a grounding mechanism for Assertion, when the agent gives the user directions, and Answer, when the agent answers the user's questions.

Generating the Response with nonverbal signals

The first job of the DM is to plan the response to a user's query. When a user asks for directions, the DM receives an event from the UM stating this intention. The Response Planner in the DM, recognizing the user's direction-request, calculates the directions, which are broken up into segments. These segments are added to the DM's Agenda, the stack of UUs to be processed. At this point, the GrM sends the first UU (a direction segment) on the Agenda to the GM to be processed. The GM converts the UU into speech and animation commands.

Table 5.3: Nonverbal signals by MACK

UU type	probability	MACK nonverbal behavior	
		within UU	pause
Assertion	0.66	keep *gM*	*gM*
	0.14	shift *gM* to *gP*	*gP*
	0.11	shift *gM* to *gP*	*gM*
	0.09	keep *gP*	*gP*
Answer	0.45	keep *gM*	*gM*
	0.36	keep *gP*	*gP*
	0.18	shift *gM* to *gP*	*gP*
Elaboration	0.47	keep *gP*	*gP*
	0.2	keep *gP*	*gM*
	0.2	keep *gM*	*gM*
	0.13	shift *gM* to *gP*	*gP*

For MACK's own nonverbal grounding acts, selection rules shown in Table 5.3 are defined based on our empirical data, and the GM determines MACK's gaze behavior by looking up the rules. For example, when MACK generates a direction segment (an Assertion), he keeps looking at the map 66% of the time. When elaborating a previous UU, he gazes at the user 47% of the time. Commands for these nonverbal acts are packed with UU (verbal content), and are sent to the GM, where these verbal and nonverbal contents are generated in a synchronized way.

Judgment of grounding by exploiting user's nonverbal signals

When MACK finishes uttering a UU, the Grounding Module (GrM) judges whether or not the UU is grounded, based on the user's verbal and nonverbal behaviors during and after the UU.

Using verbal evidence: If the user returns an acknowledgement, such as "OK", the GrM judges the UU grounded. If the user explicitly reports failure in perceiving MACK's speech (e.g., "what? "), or not-understanding (e.g., "I don't understand"), the UU remains ungrounded. Note that, for the moment, verbal evidence is considered stronger than nonverbal evidence.

Using nonverbal evidence: A mechanism for nonverbal grounding process is implemented based on a model of face-to-face grounding proposed in section **??**.

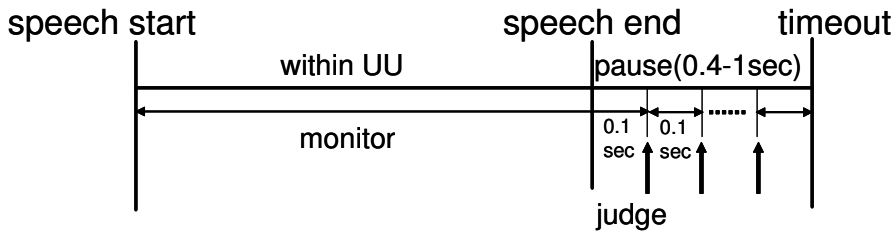


Figure 5.12: Process of grounding judgment

Table 5.4: Grounding Model for MACK

Target UU Type	Evidence Type	NV Pattern	Judgment of ground	Suggested next action
Assertion	positive	within: map pause: map /nod	grounded	go-ahead: 0.7 elaboration: 0.30
	negative	within: gaze pause: gaze	ungrounded	go-ahead: 0.27 elaboration:0.73
Answer	positive	within: gaze pause: map	grounded	go-ahead: 0.83 elaboration: 0.17
	negative	pause: gaze	ungrounded	go-ahead: 0.22 elaboration: 0.78

The following steps describe how each step in the model is implemented in MACK. The process is illustrated in Figure 5.12.

Step 1: Preparing for the next UU By referring to the Agenda, the DM can identify the speech act type of the next UU, which is the key to specify positive/negative evidence in grounding judgment later.

Step 2: Monitoring The GrM sends the next UU to GM (in Section 5.5.3), and the GM begins to process the UU. At this time, the GM logs the start time in the Discourse Model. When it finishes processing (as it sends the final command to the animation module), it logs the end time. The GrM waits for this speech and animation to end (by polling the Discourse Model until the end time is available), at which point it retrieves the timing data for the UU, in the form of timestamps for the UU start and finish. This timing data

is used to look up the nonverbal behavior co-occurring with the utterance in order to judge whether or not the UU was grounded.

Step 3: Judging When the GrM receives end signal from the GM, it starts judgment of grounding. The GrM looks up the nonverbal behavior occurring during the utterance, and compares it to the model shown in Table 5.4. For each type of speech act, this model specifies the nonverbal behaviors that signal positive or explicit negative evidence that were found in Section 5.3.3. As a first trial, the nonverbal behaviors within the UU and for the first tenth of second of a pause are used for the judgment. If these two behaviors ("within" and "pause") match the positive evidence pattern for a given speech act, then the GrM judges that the UU has been grounded. If they match a pattern for negative evidence, the UU is not grounded. If no pattern is matched during the first tenth of second of a pause, then MACK monitors the user's nonverbal behaviors for another one tenth of second, and judges again. The GrM continues looping in this manner until the UU is either grounded or ungrounded explicitly, or the timeout, which is one second into a pause. If the GrM times out without a decision, it judges the UU ungrounded.

Updating the Discourse State

After judging grounding, the GrM updates the Discourse Model. The Discourse State maintained in the Discourse Model is similar to the information state in TRINDI kit (Matheson, Poesio, and Traum, 2000), except that we store nonverbal information. There are three key fields:

(1) GROUNDED: a list of grounded UUs.

(2) UNGROUNDED: a list of pending (ungrounded) UUs.

(3) CURRENT: the current UU being processed.

If the current UU (CURRENT) is judged grounded, its belief is added to GROUNDED. If ungrounded, the UU is stored in UNGROUNDED. If an UU has subsequent contributions such as elaboration, these are stored in a single discourse unit, and grounded together when the last UU is grounded.

Determining the next action after updating the discourse state

After judging the UU's grounding, the GrM decides what MACK does next according to the result of the judgment. There are three possibilities shown as follows:

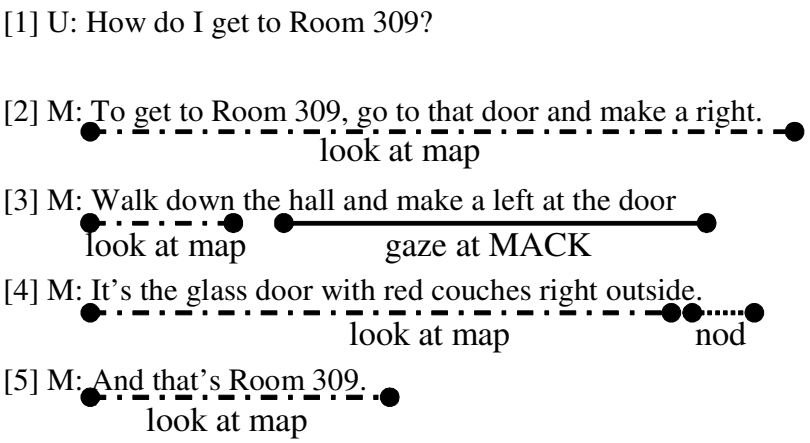


Figure 5.13: Example of user (U) interacting with MACK (M)

(Case 1) If the previous UU is successfully grounded, then continue the next UU in the Agenda. MACK can continue giving the directions as normal, by sending on the next segment in the Agenda to the GM. As shown in Table 5.4, this happens 70% of the time when the UU is grounded, and only 27% of the time when it is not grounded. Note, this happens 100% of the time if verbal acknowledgement (e.g. "Uh huh") is received for the UU.

(Case 2) If the previous UU has not been grounded yet, elaborate the previous UU. MACK can elaborate on the most recent stage of the directions. Elaborations are generated 73% of the time when an Assertion is judged ungrounded, and 78% of the time for an ungrounded Answer. MACK elaborates by describing the most recent landmark in more detail. For example, if the directions were "Go down the hall and make a right at the door, " he might elaborate by saying "The big blue door. " In this case, the GrM asks the Response Planner (RP) to provide an elaboration for the current UU; the RP generates this elaboration (looking up the landmark in the database) and adds it to the front of the Agenda; and the GrM sends this new UU on to the GM.

(Case 3) If the user gives MACK explicit verbal evidence of not understanding, MACK will simply repeat the last thing he said, by sending the UU back to the GM.

5.5.4 Example

Figure 5.13 shows an example of a user's interaction with MACK. A snapshot is shown in Figure 5.14. The user asks MACK for directions, and MACK replies using

Figure 5.14: MACK with user

speech and pointing (using a projector) to the shared map.

When the GrM sends the first segment in the Agenda to the GM, the starting time of the UU is noted and it is sent to the AM to be spoken and animated. During this time, the user's nonverbal signals are logged in the Discourse Model. When the UU has finished, the GrM evaluates the log of the UU and of the very beginning of the pause (by waiting a tenth of a second and then checking the nonverbal history). In this case, MACK noted that the user looked at the map during the UU[2], and continued to do so just afterwards. This pattern matches the positive evidence for Assertion. The UU is judged as grounded, and the grounded belief is added to the Discourse Model.

MACK then utters the second segment, UU[3], as before, but this time the GrM finds that the user was looking up at MACK during most of the UU as well as after it, which signals that the UU is not grounded. Therefore, the Response Planner generates an elaboration in UU[4]. This UU is judged to be grounded both because the user continues looking at the map, and because the user nods, and so the final stage of the directions is spoken. This is also grounded, leaving MACK ready for a new inquiry.

5.6 Preliminary Evaluation

Although we have shown an empirical basis for our implementation, it is important to ensure both that human users interact with MACK as we expect, and that their interaction is more effective than without nonverbal grounding. The issue of effec-

Table 5.5: Preliminary Evaluation

		with-grounding	w/o-grounding
num of UUs		5	4
Shift to	*gMgM*	3	2
	gPgM	2	0
	gMgP	1	0
	gPgP	1	0
	gMgMwN	0	1
	total	7	3

tiveness merits a full-scale study and thus we have chosen to concentrate here on whether MACK elicits the same behaviors from users as does interaction with other humans.

5.6.1 Procedure

Two subjects were assigned to one of the following two conditions, both of which were run as Wizard of Oz (that is, "speech recognition" was carried out by an experimenter):

(a) MACK-with-grounding: MACK recognized user's nonverbal signals for grounding, and displayed his nonverbal signals as a speaker.

(b) MACK-without-grounding: MACK paid no attention to the user's nonverbal behavior, and did not display nonverbal signals as a speaker. He gave the directions in one single turn.

Subjects were instructed to ask for directions to two places, and were told that they would have to lead the experimenters to those locations to test their comprehension. We analyzed the second direction-giving interaction, after subjects became accustomed to the system.

5.6.2 Results

In neither condition, did users return verbal feedback during MACK's direction giving. As shown in Table 5.5, in MACK-with-grounding 7 nonverbal status transitions

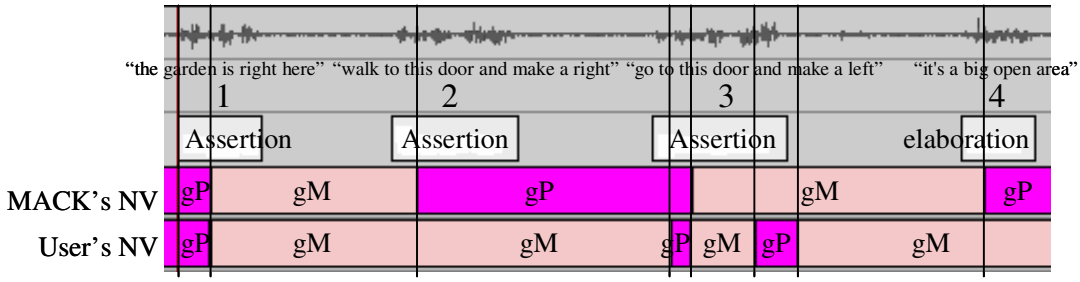


Figure 5.15: Interaction log for user-MACK conversation

were observed during his direction giving, which consisted of 5 Assertion UUs, and one of them is an elaboration.

The details of part of this interaction are shown in Figure 5.15. At the beginning of the interaction, [1], the user and MACK look at each other (gP/gP). MACK starts speaking and looks at the map, then the user follows MACK's gaze and looks at the map (gM/gM). In [2], MACK looks at the user, but the user keeps looking at the map (gP/gM), which is positive evidence of understanding, and so MACK continues on. However, in [3], the user looks at MACK and keeps looking after MACK's speech has finished (gM/gP). This shows negative evidence of understanding. So, for the next UU in [4], MACK elaborates the previous UU. The user then looks back at the map, and MACK continues on to the next segment of the directions.

5.6.3 Discussion

As demonstrated in the example, when MACK used nonverbal grounding, the transition patterns between MACK and the user are strikingly similar to those in our empirical study of human-to-human communication. There were three transitions to gM/gM (both look at the map), which is a normal status in map task conversation, and two transitions to gP/gM (MACK looks at the user, and the user looks at the map), which is the most frequent transition in Assertion as reported in Section 3. Moreover, in MACK's third UU, the user began looking at MACK at the middle of the UU and kept looking at him after the UU ended. This behavior successfully elicited MACK's elaboration in the next UU.

On the other hand, in the MACK-without-grounding condition, the user never looked at MACK, and nodded only once, early on. As shown in Table 5.5, only three transitions were observed (shift to gMgM at the beginning of the interaction, shift to gMgMwN, then back to gMgM).

While a larger scale evaluation with quantitative data is one of the most important issues for future work, the results of this preliminary study strongly support our model, and show MACK's potential for interacting with a human user using

human-human conversational protocols.

5.7 Summary and Discussion

This chapter has shed light on nonverbal behaviors in face-to-face grounding, presenting both an empirical study of human face-to-face communication and implementation of an embodied conversational agent. First, we reported how people use nonverbal signals in the process of grounding. We found that nonverbal signals recognized as positive evidence of understanding are different depending on the type of speech act. We also found that a listener's gaze at the shared map serves as positive evidence of understanding typically in Assertion. As mentioned in Section 5.2, previous studies of human communication reported that the frequency of gaze at the partner decreases when conversational participants engage in a task requiring joint attention to a shared reference (Anderson et al., 1997; Argyle and Graham, 1977; Whittaker and O'Conaill, 1993) Our empirical study supports these works and gives more detailed analysis in terms of grounding, describing when and with what types of speech act gazing at the shared objects frequently occurs. We also found that maintaining gaze on the speaker even after the speakers' utterance has finished is interpreted as evidence of not-understanding, evoking additional information from the speaker.

Based on these empirical results, we proposed a model of face-to-face grounding consisting of three steps; Prepare, Monitor, and Judge. The advantage of this model over previous ones is to allow updating the common ground without verbal feedback. Employing nonverbal information as evidence of understanding/not-understanding, and utterance unit as unit of grounding, we provided a more fine-grained process model of face-to-face grounding.

Then, we implemented the model into our embodied conversational agent, MACK. It can recognize different kinds of user's nonverbal information, such as gaze-direction and head nod, and exploit them in updating the discourse state. In addition, MACK also can generate appropriate nonverbal signals according to the type of speech act that he is speaking. The results of preliminary evaluation strongly support our model, and show MACK's potential for interacting with a human user employing human-human conversational protocols.

Although our model of face-to-face grounding can handle more fine-grained processes of grounding than previous models, it still needs to be improved. One important aspect of grounding which our model cannot account for is that the grounding criterion should change depending on the purpose of conversation, as reviewed in section 2.1 (Chapter 2) . Updating common ground through the grounding process contributes to maintaining conversation to prevent communication failure. Therefore, it is often not necessary to assure perfect understanding of each contribution,

but only understanding "to a criterion sufficient for current purposes" (Clark and Wilkes-Gibbs, 1986).

As described in Chapter 2, distinguishing different levels of conversation is important in modeling human communication, and also very useful in maintaining human-computer interaction. User's voice may or may not be recognized as speech sound, an out-of-grammar utterance, an utterance whose meaning is ambiguous, or an utterance whose communicative act cannot be accepted. In the current implementation, MACK simply repeats the last thing he said when the user gives verbal evidence of not understanding, like "I don't understand". This is because the system cannot deal with levels of conversation. If the Grounding Module can judge the level of current discourse, and add this information to the Discourse State, the Dialogue Manager can detect the cause of miscommunication by referring to the Discourse State, and change the response depending on the levels of miscommunication. In order to deal with such uncertainty in grounding, incorporating a probabilistic approach (Paek and Horvitz, 1999) into our model of face-to-face grounding is an elegant possibility.

Second, it is also an important future direction to make the model more comprehensive. While we focused on eye gaze and head nods, which directly contribute to grounding, other types of nonverbal behaviors collateral with speech need to be incorporated into the model. They would specifically contribute to establishing a shared reference when people are engaged in a task requiring joint attention to complex objects. Clark (2003) claimed that communication is ordinarily anchored to the material world. He proposed "Directing-to" and "Placing-for" as techniques for indicating. Directing-to is a speaker's signal that directs addressee's attention to object *o*. Placing-for is a speaker's signal that places object o for addressee's attention. Both of these are techniques used to connect a message and the physical world that the message describes, and get the addressee accessible and perceivable to the message. As devices for directing-to, Clark (2003) listed various kinds of non-verbal behaviors using different body parts. For example, pointing gesture is a powerful device for directing listeners' attention to a reference.

Moreover, we have noticed that the usage of nonverbal behaviors is not always consistent, and sometimes contradictions occur between verbal and nonverbal evidence (e.g., an interlocutor says, "OK", but looks at the partner). In addition, eye-gaze and head nod may serve multiple functions at one time. In terms of floor management, gazing at the partner is a signal of giving up a turn (Duncan, 1974), suggesting that listeners are trying to elicit more information from the speaker. Thus, speakers gaze at listeners at the end of Information Request participates in grounding process as well as floor management (giving up speakers' turn). The model needs to be extended to handle these complex cases.

Finally, although our model is based on empirical findings and supported by

clear results in our preliminary evaluation, we admit that a larger scale evaluation with various measures needs to be conducted. The evaluation would be concerned with the task performance, verbal and non-verbal characteristics of interaction, and a subjective evaluation using the following criteria: agent's language understanding/use, smoothness of interaction, lifelikeness, social attractiveness, and trustworthiness (Cassell and Thorisson, 1999; Nass, Isbister, and Lee, 2000).

Chapter 6

Generating Gestures for Presentation Agents

As a contribution to the *Generation Module*, this chapter proposes CAST, a mechanism that automatically generates gestures for a conversational agent. In addition, as an application of the CAST mechanism, this chapter presents a web-based multimedia environment, SPOC (Stream-oriented Public Opinion Channel), which allows novice users to embody a story as a multimodal presentation, and distribute it on the network. The system produces a digital camera work for graphics and video clips, and generates agent animations automatically using CAST. These mechanisms allow users to create multimedia contents featuring agent animations very easily.

In the next two sections, we describe the background of this study. Section 6.3, proposes the SPOC mechanisms, showing a process of how these mechanisms embody a story as a multimedia content. In Section 6.5, first we conduct an empirical study to establish a model of assigning gestures to text. Our empirical study identifies lexical and syntactic information strongly correlated with gesture occurrence, and suggests that syntactic structure is more useful for judging gesture occurrence than local syntactic cues. Then, based on the empirical results, we implement the CAST system that converts text into a conversational agent gesticulating and speaking synchronously.

In addition to the empirical study for modeling human gesture usage, we also conduct an evaluation experiment for the whole SPOC system. The results show that SPOC is easy-to-use and easy-to-learn for novice users, suggesting that this system reduces user's cost in making a multimedia content, and encourages communication in a network community. Section 6.8 reports the evaluation experiments. Then, Section 6.9 describes relevant previous work. Finally, we summarize the contribution of this work, and discuss our future direction.

6.1 Problem

Multimedia content, such as movies and TV news stories, used to be produced solely by the corporate mass media. Thanks to the significant advances in media and network technologies over the last decade, multimedia equipment (e.g., digital still/video cameras) and Internet access have become available for personal use. This has allowed ordinary people to express their own stories as multimedia content and distribute it on the Internet.

It is still not very easy, however, for non-expert users to produce their own multimedia content. In embodying their stories as multimedia content, users need to learn how to edit multimedia materials, such as graphics, video clips, audio, and animation. For example, previous studies suggest that synchronized speech and agent animations make learning activities more effective (Craig and Gholson, 2002; van Mulken, Andre, and Mfiller, 1998), but it is almost impossible for ordinary users to create detailed designs for agent animations to be synchronized with speech. In addition, they also need to have the skill to set up streaming of the resulting content to distribute it on a network. Even for expert users, these tasks take enormous effort and time.

Our research goal is to provide an all-in-one web-based application enabling users to easily create and distribute multimedia narrative content so as to facilitate story-based communication within a network community. To accomplish this goal, we propose a multimedia environment, called SPOC (Stream-oriented Public Opinion Channel), and an animated agent system, called CAST (Conversational Agent System for neTwork applications).

First, SPOC is a server system providing the following functions: (1) automatic generation of multimedia story content by integrating speech, graphics, video clips, and agent animations; (2) broadcasting of the contents on a network; and (3) display of such multimedia content on a web browser.

CAST, working as a component of SPOC, creates a storyteller or presenter agent in SPOC. It determines the agent's nonverbal behaviors automatically according to linguistic information in a text. Because of its embodied representation, the animated agent can display nonverbal behaviors (e.g., gestures, eye movements) with its face and body. Therefore, the animated presenter agent is capable of utilizing the same communication modalities as a human in face-to-face communication.

6.2 Background

Stories can transfer tacit knowledge and help people understand a collection of events as a coherent unit. One study of cognitive psychology showed that the bulk of human knowledge and memory is communicated and encoded in story form (Schank

and Abelson, 1995). As stories seem to play a central role in human memory by providing an organizing structure for new experiences and knowledge, storytelling has been studied in a number of disciplines, including linguistics, psychology, artificial intelligence, human-computer interaction, learning environments, and knowledge management (Bruner, 1990; Aylett, 1999; Mott, Challaway, and Zettlemoyer, 1999; Lawrence and Thomas, 1999; Mateas and Stern, 2000).

On the basis of these previous studies, we have proposed the concept of "Social Intelligence Design" for a network community (Nishida, 2002). Social intelligence design employs story-based communication and conversation to establish mutual understanding and create knowledge in a society (Isaacs, 1996). To support the process of evolving and circulating social intelligence, we have already developed some web applications. Public Opinion Channel (POC) is a participatory broadcasting system that broadcasts questions, opinions, and discussions arising in a community (Fukuhara et al., 2003). This system has the following functions to support knowledge management in a community: (1) interaction with the system by posting messages and stories; (2) viewing of conversational presentations by two embodied agents; and (3) use of a "Knowledge Card", consisting of a short text (a few sentences) and a graphic image, as the information unit distributed to the community; and (4) collection and classification of information by using keywords contained in a Knowledge Card.

As an extension of the POC system, our next system, EgoChat (Kubota, Kurohashi, and Nishida, 2002), employs an agent-based approach to facilitate the process of circulating conversational information within a community. Previous studies suggest that animated avatar agents can play an important role in communication in a network community. Avatar agents facilitate seamless communication in video-mediated communication (VMC) (Nakanishi et al., 1996), as well as encourage communication in collaboration systems (Takahashi and Takeda, 2001). On the basis of these works, EgoChat enables personalized, peer-to-peer asynchronous communication. In this system, an embodied agent acting as a virtualized ego talks on a user's behalf. The user can have a conversation with any virtualized ego on the system by using a speech interface. This motivates users to enjoy interaction with the system. In addition, EgoChat supports an information-circulating process within a community by helping the users to generate, improve, integrate, and delete Knowledge Cards.

Our basic idea of story-based communication for network communities has been successfully implemented in these systems. The content generated by these systems is quite limited in its expressiveness, however, since the presentation content consists of a static graphical image and agent animations with a limited range of actions. To improve the expressiveness of content, first, more dynamic and lively visual materials are preferred, because it is hard to keep an audience's attention with static visual

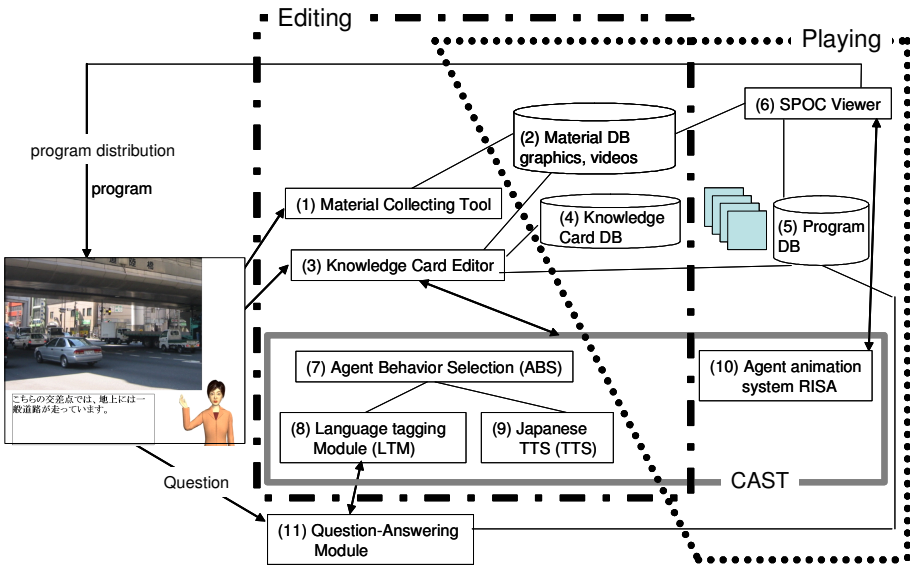


Figure 6.1: SPOC functions and components

content (Kraft, 1986). Second, our previous animated agents were not capable of displaying nonverbal communicative behaviors (Cassell et al., 2001). Even when graphics or video clips contain necessary and sufficient information, the presence of a lively speaker presenting well-prepared contents is much more appealing (Andre et al., 2000). To enable an animated agent to perform meaningful actions as a presenter, a more sophisticated agent system is necessary. Addressing these issues, we designed our new system, SPOC, by focusing on the following capabilities:

1. Streamed video clips are available as visual materials.
2. Camera work, such as zoom and pan, is automatically applied to visual materials, including both graphics and video clips.
3. The gestures and facial expressions of an animated agent are automatically selected and generated.

The following sections describe the details of SPOC's implementation and show how this system exceeds the capabilities of previous systems.

6.3 SPOC

The SPOC components and its functions are illustrated in Figure 6.1. SPOC users can (a) edit a SPOC program with the Knowledge Card Editor, (b) play a program

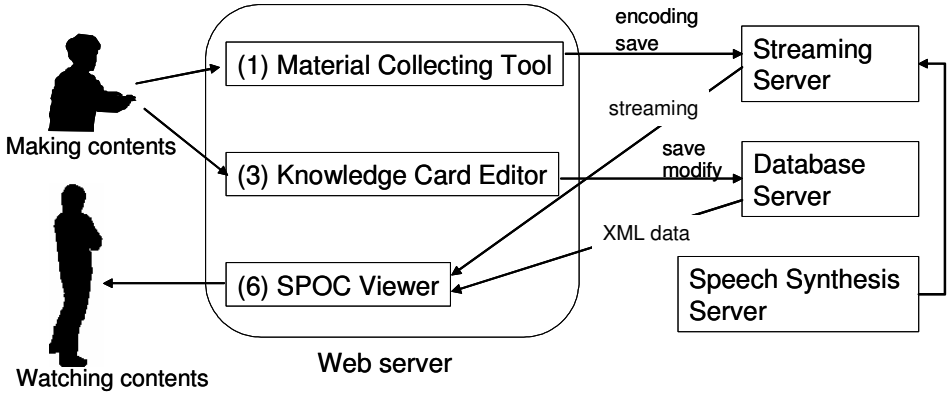


Figure 6.2: SPOC system overview

through the SPOC Viewer, and (c) ask the system a question. User can post questions while watching a program. If the Question-Answering (QA) Module (Kiyota, Kurohashi, and Kido, 2002) finds an answer for a question, it sends a SPOC program with the answer back to the Viewer, and the program is played on the Viewer. In the following subsections, in addition to describing the system architecture and the Knowledge Cards, we focus on the (a) editing and (b) playback functions.

6.3.1 SPOC System Architecture

To provide an environment for creating, editing, posting, and playing multimedia content without any software installation, all the system components run on the server side. Figure 6.2 shows an overview of the SPOC architecture. It consists of a web server and three back-end servers: the database server, the streaming server, and the speech synthesis server.

The web server, using session management, provides three web applications to the users: the Material Collecting Tool, the Knowledge Card Editor, and the SPOC Viewer. Each is connected to the back-end servers. The streaming server converts and saves video clips, which are uploaded by users through the Material Collecting Tool, as streaming data. It also serves streams of video clips and speech sounds to the users through the SPOC Viewer. The database server maintains XML data for constructing multimedia presentations, which are created and modified by the users through the Knowledge Card Editor. The speech synthesis server creates audio files for agent speech by accessing a text-to-speech engine.

Employing this system architecture, SPOC enables users to enjoy a web-based multimedia content service without installing any software, such as a multimedia authoring tool, animation software, or a text-to-speech engine.

Figure 6.3: SPOC program editing window

6.4 Editing SPOC Contents

In SPOC, a user's story is embodied as a SPOC program, which is like a TV program. Users can create their own programs by using the Knowledge Card Editor (Editor), and they can post programs on the web. Visual materials, such as graphics and video clips, are uploaded and encoded through the Material Collecting Tool and then stored in the streaming server.

A SPOC program consists of a sequence of Knowledge Cards (Cards), as shown in Figure 6.3, showing the program editing window. Each Card is like a scene in a TV program, and a user edits Cards one by one. Thus, a Card is a building block for composing a story. Users can create different stories by changing the order of the Cards. A snapshot of the Editor is shown in Figure 6.4. In editing a Card, a user only needs to do the following two things:

(I) Edit visual materials by first selecting a file from a menu, and then specifying the zoom scale and the position of the focused area. For example, in Figure 6.4, the user has focused and zoomed in on Target A in the Card. The user can do this procedure intuitively by manipulating a GUI (zoom bar). If the selected material is a movie file, the user can extract part of the video clip by specifying the start and end frames in the video.

(II) Type the text to be uttered by the animated agent.

Step (I) specifies the camera work, which is automatically generated by the SPOC Viewer, and step (II) triggers automatic script generation by CAST for the agent behavior. These steps are described in detail in Section 6.7.1 and 6.5 respectively.

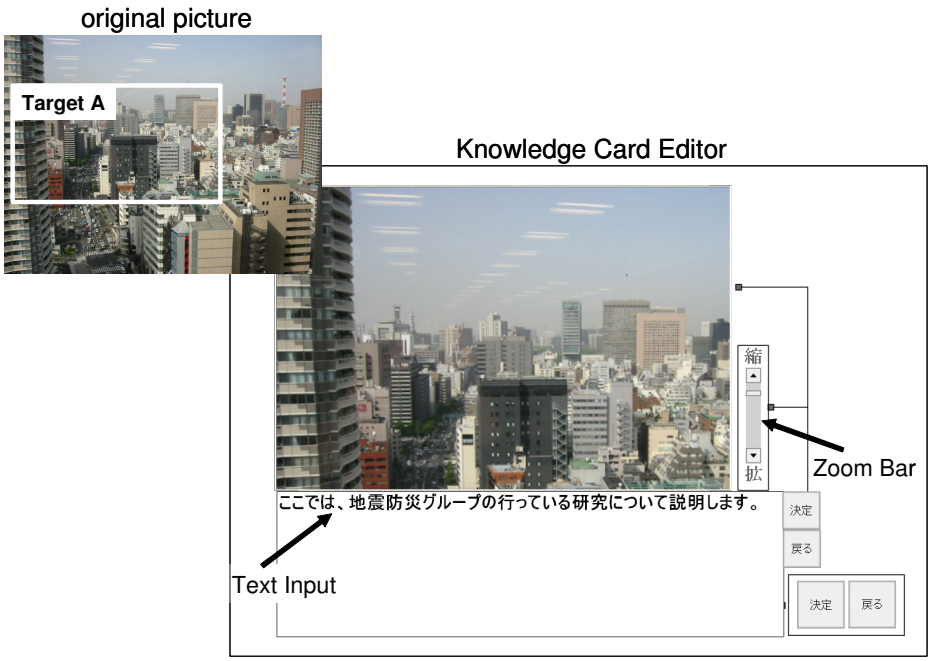


Figure 6.4: Knowledge Card Editor

6.5 CAST

This subsection describes an animated agent system, CAST. It is triggered by the Editor to calculate an agent animation schedule and produce a synthesized voice for the agent.

The next section reviews theoretical issues about the relationships between gestures and syntactic information. The empirical study we conducted based on these issues is described in Section 6.5.3. In Section 6.5.4 we describe the implementation of the CAST system.

6.5.1 Background

Previous studies in human communication suggest that gestures in particular contribute to better understanding of speech. About 90% of all gestures by speakers occur when the speaker is actually uttering something (McNeill, 1992). Experimental studies have shown that spoken sentences are heard twice as accurately when they are presented along with a gesture (Berger and Popelka, 1971). Comprehension of a description accompanied by gestures is better than that accompanied by only the speaker's face and lip movements (Rogers, 1978).

These previous studies suggest that generating appropriate gestures synchronized with speech is a promising approach to improving the performance of interface

agents. In previous studies of multimodal gen-eration, gestures were determined according to the instruction content (André, Rist, and Muller, 1999; Rickel and Johnson, 1999), the task situation in a learning environment (Lester, Stone, and Stelling, 1999), or the agent's communicative goal in conversation (Cassell, Stone, and Yan, 2000). These approaches, however, require the contents developer (e.g., a school teacher designing teaching materials) to be skilled at describing semantic and pragmatic relations in logical form.

A different approach, Cassell, Vilhjalmsson, and Bickmore (2001) proposes a toolkit that takes plain text as input and automatically suggests a sequence of agent behaviors synchronized with the synthesized speech. However, there has been little work in computational linguistics on how to identify and extract linguistic information in text in order to generate gestures.

Our study has addressed these issues by considering two questions. (1) Is the lexical and syntactic information in text useful for generating meaningful gestures? (2) If so, how can the information be extracted from the text and exploited in a gesture decision mechanism in an interface agent? Our goal is to develop a media conversion technique that generates agent animations synchronized with speech from plain text.

6.5.2 Linguistic Theories and Gesture Studies

In this section, we review linguistic theories and discuss the relationship between gesture occurrence and syntactic information.

Linguistic quantity for reference: McNeill (1992) used communicative dynamism (CD), which represents the extent to which the message at a given point is "pushing the communication forward" (Firbas, 1971), as a variable that correlates with gesture occurrence. The greater the CD, the more probable the occurrence of a gesture. As a measure of CD, McNeill chose the amount of linguistic material used to make the reference (Givon, 1985). Pronouns have less CD than full nominal phrases (NPs), which have less CD than modified full NPs. This implies that the CD can be estimated by looking at the syntactic structure of a sentence.

Theme/Rheme: McNeill also asserted that the theme (Halliday, 1967) of a sentence usually has the least CD and is not normally accompanied by a gesture. Gestures usually accompany the rhemes, which are the elements of a sentence that plausibly contribute information about the theme, and thus have greater CD. In Japanese grammar there is a device for marking the theme explicitly. Topic marking postpositions (or "topic markers"), typically "wa," mark

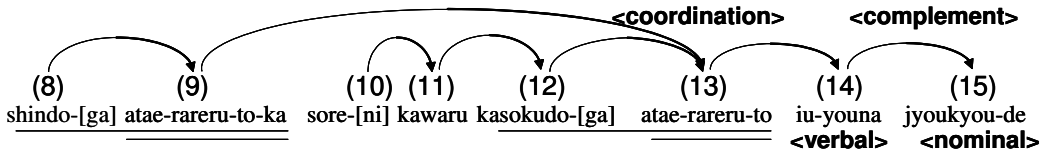


Figure 6.5: Example analysis of syntactic dependency. Underlined phrases are accompanied by gestures, and strokes occur at double-underlined parts. Case markers are enclosed by square brackets [ ].

a nominal phrase as the theme. This facilitates the use of syntactic analysis to identify the theme of a sentence.

Another interesting aspect of information structure is that in English grammar, a wh-interrogative (what, how, etc.) at the beginning of a sentence marks the theme and indicates that the content of the theme is the focus (Halliday, 1967). However, we do not know whether such a special type of theme is more likely to co-occur with a gesture or not.

Given/New: Given and new information demonstrate an aspect of theme and rheme. Given information usually has a low degree of rhematicity, while new information has a high degree. This implies that rhematicity can be estimated by determining whether the NP is the first mention (i.e., new information) or has already been mentioned (i.e., old or given information).

Contrastive relationship: Prevost (1996) reported that intonational accent is often used to mark an explicit contrast among the salient discourse entities. On the basis of this finding and Kendon's theory about the relationship between intonation phrases and gesture placements (Kendon, 1972), Cassell and Prevost (1996) developed a method for generating contrastive gestures from a semantic representation. In syntactic analysis, a contrastive relation is usually expressed as a coordination, which is a syntactic structure including at least two conjuncts linked by a conjunction.

Figure 6.5 shows an example of the correlation between gesture occurrence and the dependency structure of a Japanese sentence. Bunsetsu units (8)-(9) and (10)-(13) in the figure are conjuncts. A "bunsetsu unit" in Japanese corresponds to a phrase in English, such as a noun phrase or a prepositional phrase. Each conjunct is accompanied by a gesture. Bunsetsu (14) is a complement containing a verbal phrase; it depends on bunsetsu (15), which is an NP. Thus, bunsetsu (15) is a modified full NP and thus has large linguistic quantity.

6.5.3 Empirical Study

To identify linguistic features that might be useful for judging gesture occurrence, we videotaped seven presentation talks and transcribed three minutes for each of them. The collected data included 2124 bunsetsu units and 343 gestures.

Gesture Annotation

Three coders discussed how to code the half the data and reached a consensus on gesture occurrence. After this consensus on the coding scheme was established, one of the coders annotated the rest of the data. A gesture consists of preparation, stroke, and retraction (McNeill, 1992), and a stroke co-occurs with the most prominent syllable (Kendon, 1972). Thus, we annotated the stroke time as well as the start and end time of each gesture.

Linguistic Analysis

Each bunsetsu unit was automatically annotated with linguistic information using a Japanese syntactic analyzer (Kurohashi and Nagao, 1994). The information was determined by asked the following questions for each bunsetsu unit.

(a) If it is an NP, is it modified by a clause or a complement?

(b) If it is an NP, what type of postpositional particle marks its end (e.g., "wa", "ga", "wo")?

(c) Is it a wh-interrogative?

(d) Are all the content words in the bunsetsu unit have mentioned in a preceding sentence?

(e) Is it a constituent of a coordination?

Moreover, as we noticed that some lexical entities frequently co-occurred with a gesture in our data, we used the syntactic analyzer to annotate additional lexical information based on the following questions.

(f) Is the bunsetsu unit an emphatic adverbial phrase (e.g., very, extremely), or is it modified by a preceding emphatic adverb (e.g., very important is-sue)?

(g) Does it include a cue word (e.g., now, therefore)?

(h) Does it include a numeral (e.g., thousands of people, 99 times)? We then investigated the correlation between these lexical and syntactic features and the occurrence of gesture strokes.

Table 6.1: Summary of results

Case ID	Case			Frequency per bunsetsu unit
[C1]	Quantity of modification	(a) NP modified by a clause		0.382
[C2]		Pronouns, other type of NPs	(b) Case marker = "wo" & (d) New information	0.281
[C3]	(c) WH-interrogative			0.414
[C4]	(e) Coordination			0.477
[C5]	Emphatic adverb	(f) Emphatic adverb itself		0.244
[C6]		(f') Following an emphatic adverb		0.350
[C7]	(g) Cue word			0.415
[C8]	(h) Numeral			0.393
[C9]	Other (baseline)			0.101
[C10]	(i) Demonstrative			deictic gesture

Result

The results are summarized in Table 6.1. The baseline gesture occurrence frequency was 10.1% per bunsetsu unit (a gesture occurred once about every ten bunsetsu units). A gesture stroke most frequently co-occurred with a bunsetsu unit forming a coordination (47.7%). When an NP was modified by a full clause, it was accompanied by a gesture 38.2% of the time.

For the other types of noun phrases, including pronouns, when an accusative case marked with case marker "wo" was new information (i.e., it was not mentioned in a previous sentence), a gesture co-occurred with the phrase 28.1% of the time.

Moreover, gesture strokes frequently co-occurred with wh-interrogatives (41.4%), cue words (41.5%), and numeral words (39.3%). Gesture strokes frequently occurred right after emphatic adverbs (35%) rather than with the adverb (24.4%).

These cases listed in Table 6.1 had a 3 to 5 times higher probability of gesture occurrence than the baseline and accounted for 75% of all the gestures observed in the data. Our results suggest that these types of lexical and syntactic information can be used to distinguish between where a gesture should be assigned and where one should not be assigned. They also indicate that the syntactic structure of a sentence more strongly affects gesture occurrence than theme or rheme and than

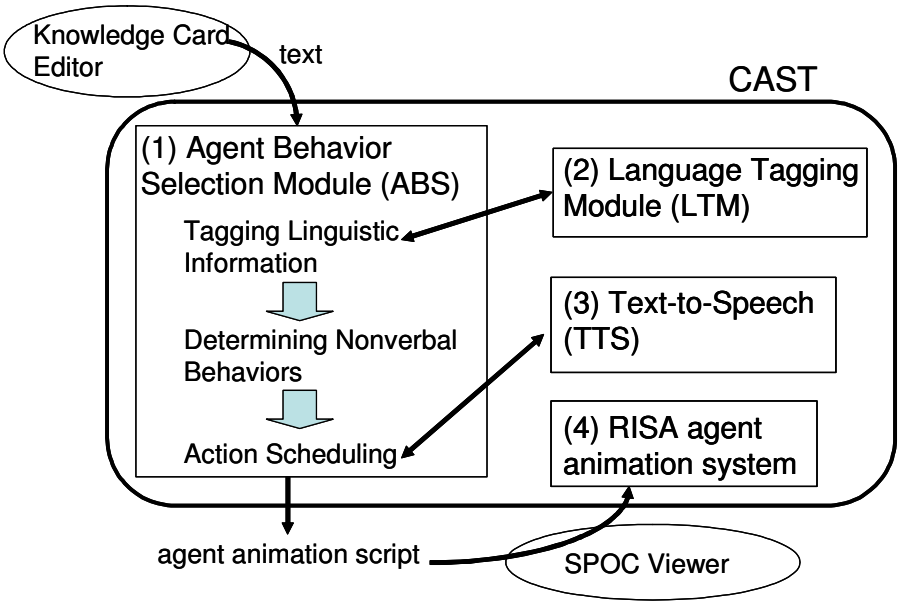


Figure 6.6: CAST architecture

given or new information specified by local grammatical cues, such as topic markers and case markers.

6.5.4 System Implementation

Overview

We used our results to build a presentation agent system, SPOC (Stream-oriented Public Opinion Channel). This system enables a user to embody a story (written text) as a multimodal presentation featuring video, graphics, speech, and character animation.

In order to implement a storyteller in SPOC, we developed an agent behavior generation system we call CAST. Taking text input, CAST automatically selects agent gestures and other nonverbal behaviors, calculates an animation schedule, and produces synthesized voice output for the agent.

As shown in Figure 6.6 and 6.7, CAST consists of four main modules: (1) the Agent Behavior Selection Module (ABS), (2) the Language Tagging Module (LTM), (3) a Text-to-Speech engine (TTS), and (4) a Flash-based character animation system, RISA (RIStex animated Agent system). When CAST receives a text input, it sends the text to the ABS. The ABS selects appropriate gestures and facial expressions according to linguistic information calculated by the LTM. Then, the ABS obtains timing information by accessing the TTS, and it calculates a time schedule for the set of agent actions. The output from the ABS is a set of animation

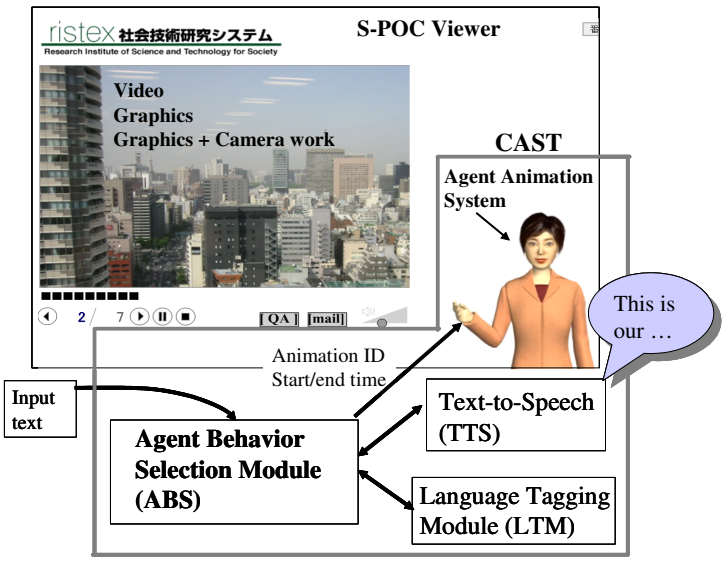


Figure 6.7: Overview of CAST and SPOC

instructions that can be interpreted and executed by the RISA animation system.

Determining Agent Behaviors

Tagging linguistic information First, the LTM parses the input text and calculates the linguistic information described in Section 6.5.2. For example, bunsetsu (9) in Figure 6.5 has the following feature set.

> *Text-ID: 1, Sentence-ID: 1, Bunsetsu-ID: 9, Govern: 8, Dependon: 13, Phrase-type: VP, Linguistic-quantity: NA, Case-marker: NA, WH-interrogative: false, Given/New: new, Coordinate-with: 13, Emphatic-Adv: false, Cue-Word: false, Numeral: false*

The text ID of this bunsetsu unit is 1, the sentence ID is 1, the bunsetsu ID is 9. This bunsetsu governs bunsetsu 8 and depends on bunsetsu 13. It conveys new information and, together with bunsetsu 13, forms a parallel phrase.

Assigning gestures Then, for each bunsetsu unit, the ABS decides whether to assign a gesture or not based on the empirical results shown in Table 6.1. For example, bunsetsu unit (9) shown above matches case C4 in Table 6.1, where a bunsetsu unit is a constituent of coordination.

In this case, the system assigns a gesture to the bunsetsu with 47.7% probability. In the current implementation, if a specific gesture for an emphasized concept is

```
[1] shindo-ga
        <Gesture_right type="contrast" handshape_right="stroke1@2">
[2] atae-rareru-to-ka
        </Gesture_right>
[3] sore-ni
[4] kawaru
[5] kasokudo-ga
        <Gesture_right type="contrast" handshape_right="stroke2@2">
[6] atae-rareru-to
        </Gesture_right>
[7] iu-youna
        <Gesture_right type="best" handshape_right="stroke1">
[8] jyoukyou-de
        </Gesture_right>
    ...
```

Figure 6.8: Example of CAST output

defined in the gesture animation library (e.g., a gesture animation expressing "big"), it is preferred to a "beat gesture" (a simple flick of the hand or fingers up and down (McNeill, 1992)). If a specific gesture is not defined, a beat gesture is used as the default.

The output of the ABS is stored in XML format. The type of action and the start and end times of the action are indicated by XML tags. In the example shown in Figure 6.8, the agent first gazes towards the user. It then performs contrast gestures at the second and sixth bunsetsu units and a beat gesture at the eighth bunsetsu unit.

Finally, the ABS transforms the XML into a time schedule by accessing the TTS engine and estimating the phoneme and bunsetsu boundary timings. The scheduling technique is similar to that described by (Cassell, Vilhjalmsson, and Bickmore, 2001). The ABS also assigns visemes for the lip-sync and the facial expressions, such as head movement, eye gaze, blink, and eyebrow movement.

Action Scheduling

After determining the nonverbal behaviors, the next step is to generate a time schedule to be executed by the animation system. To synchronize an agent's speech with nonverbal behaviors, the Scheduling Module in the ABS accesses the TTS engine to obtain the timing information for each phoneme (phoneme type, start time, and duration) and the bunsetsu boundaries. At this point, a synthesized voice

produced by the TTS engine is saved in the streaming server. A viseme [1]@for the lip-sync process is assigned according to the phoneme type. The output of the Scheduling Module is formatted as a set of instructions to be executed by the RISA animation system. Each command in the instruction set specifies an action type and the start time of the animation. An example of an instruction set is shown below:

<START AID="A669" ACTION="GESTURE_RIGHT" TYPE="DEICTIC"
HANDSHAPE_RIGHT="POINTING" SRT="2.88">
<START AID="A671" ACTION="EYEBROWS" SRT="2.88">
<START AID="A673" ACTION="VISEME" TYPE="D" SRT="2.88">
.....
<START AID="A679" ACTION="VISEME" TYPE="O" SRT="3.25">
<START AID="A680" ACTION="VISEME" TYPE="E" SRT="3.36">
<STOP AID="A671" ACTION="EYEBROWS" SRT="3.40">
<STOP AID="A669" ACTION="GESTURE_RIGHT" TYPE="DEICTIC"
HANDSHAPE_ RIGHT="POINTING" SRT="3.40">

START or STOP at the beginning of a command indicates whether the command starts or stops the action. AID indicates the action ID. The ACTION attribute specifies the type of action, such as GESTURE_RIGHT or VISEME. For VISEME, the viseme type is specified by the TYPE attribute. For example, TYPE="D" indicates that the lip shape for the "D" sound should be used. SRT specifies the time at which the command should be executed. For example, in the action AID="A669", a right-hand pointing gesture starts at 2.88 sec, and the hand returns to the original position at 3.40 sec. Finally, the animation instruction set is sent back to the Knowledge Card Editor and saved in XML format.

6.6 Structure of a Knowledge Card

Next, we reconsider the whole process of producing content by describing the structure of a Knowledge Card. As a result of the two-step Editor procedure described in Section 6.4, a Knowledge Card is automatically generated and saved in the Knowledge Card DB in XML format. An example of the XML code is shown in Figure 6.9.

<CARDS> represents the beginning of a new program. This consists of CARD elements. A <CARD> element is the building block of a program and is composed of ID, BOX, IMAGE, AGENT, and COMMENT elements. <ID> specifies the ID of the CARD. <BOX> specifies the order of the card in a program. An <IMAGE>

[1] A viseme is a generic facial image that can be used to describe a particular sound. A viseme is the visual equivalent of a phoneme or unit of sound in spoken language.

<?xml version="1.0" encoding="Shift_JIS" ?>
<cards id="**sgkc4lwPh4SEl9js**" user_id="**wall**">
<card>
<id>**sj8SWPc9mnBrS4x1**</id>
<box>**1**</box>
<image>
<imageurl>**http://000.000.00.000:8080/sgkc4lwPh4SEl9js/s1.jpg**</imageurl>
<xscale>**65.9958**</xscale>
<yscale>**65.9958**</yscale>
<xpos>**0.0**</xpos>
<ypos>**0.0**</ypos>
<inipos>**0**</inipos>
<endpos>**0**</endpos>
</image>
<AGENT SCENE="**sj8SWPc9mnBrS4x1**" UtrID="**0**">
<Action ID="**1650000**" Srt="**0.0**" />
<Action ID="**1680000**" Srt="**0.0**" />
<Action ID="**18500**" Srt="**0.0**" />
<Action ID="**188**" Srt="**0.02002**" />
<Action ID="**189**" Srt="**0.06108102000000001**" />
<Action ID="**188**" Srt="**0.13021110102000003**" />
<Action ID="**189**" Srt="**0.21342431212102**" />
<Action ID="**188**" Srt="**0.289713736433141**" />
<Action ID="**186**" Srt="**0.3240374501695742**" />
<Action ID="**185**" Srt="**0.4282236789904709**" />
<Action ID="**18500**" Srt="**0.5467699026694613**" />

.
.
.

<comment>**ここでは、地震防災グループの取り組んでいる、
事実の明示化に関する研究を紹介します。**</comment>

“I will present research on making clear the facts that the Earthquake Disaster Prevention Research Group is conducting.”

Figure 6.9: Example of a Card in XML format

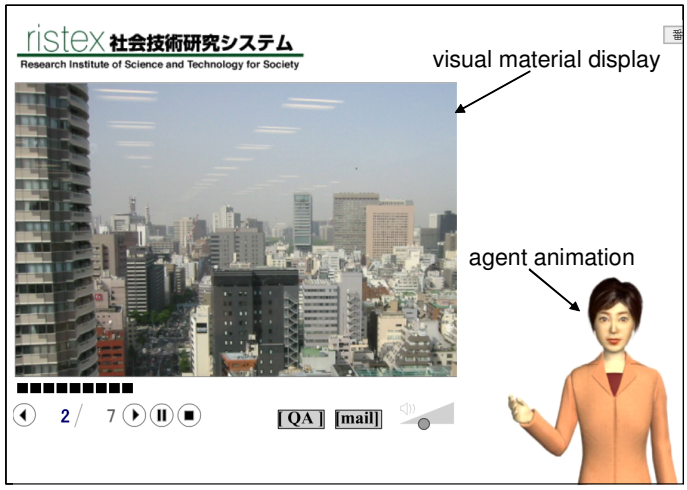


Figure 6.10: Snapshot of SPOC Viewer

element consists of several sub-elements specifying the visual material in detail. <IMAGEURL> specifies the URL address where the graphics and video clips are stored. <XSCALE> and <YSCALE> specify the horizontal and vertical zoom scales, respectively, as percentages (%). <XPOS> and <YPOS> specify the horizontal and vertical positions, respectively, of the material in a display. <INIPOS> and <ENDPOS> specify the respective start and end frames of video material. In the case of a graphic image, the value of the data is "0". The data for these tags are specified while editing the visual material (Section 6.4). An <AGENT> element contains the set of animation instructions generated by CAST (Section 6.5). Finally, <COMMENT> indicates text in the card.

When the SPOC Viewer plays a Card, it interprets the XML tags and displays all the materials according to the instructions. By repeating this process, SPOC generates a multimedia presentation from a sequence of Cards. The details of this process are described in the next subsection.

6.7 Viewing SPOC Contents

Using the SPOC Viewer (Viewer), users can watch programs posted by other community members. The input to the Viewer is a set of Knowledge Cards created by the Editor. The Viewer produces an audio-visual stream by playing each Card in a program one by one. A screen shot of the SPOC Viewer is shown in Figure 6.10. The Viewer consists of a visual material display and an animated agent. The visual material display shows a graphic image or a video clip. The agent animations are generated by the flash-based animation system RISA, described in Section 6.7.2.

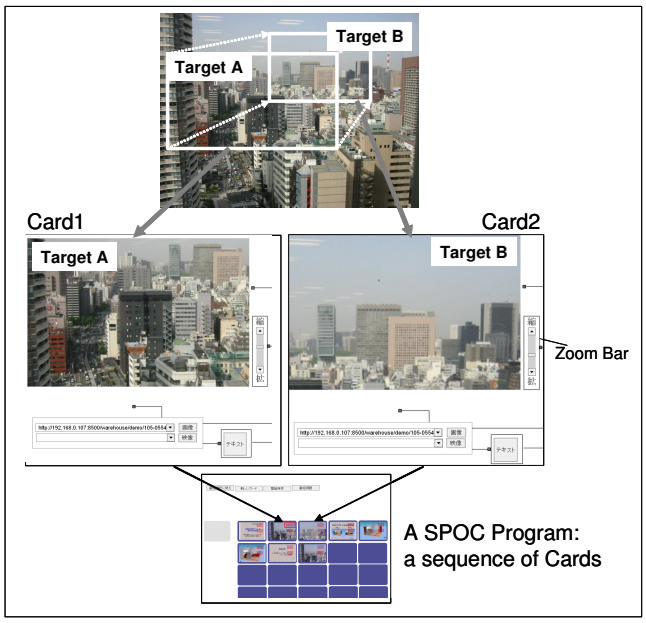


Figure 6.11: Camera work

6.7.1 Automatic Camera Work Generation

The Viewer automatically generates digital camera work. In playing a program, the Viewer compares two consecutive cards. If both cards use the same original visual material, camera work is automatically applied. The Viewer calculates the difference in the zoom scale and the focused area position between the Cards. It then gradually changes the zoom scale and the focused area from one Card to another.

For example, as shown in Figure 6.11, Cards 1 and 2 use the same image. Card 1 is focused on Target A. This information is saved in a Knowledge Card XML:
<XSCALE>150</XSCALE>
<YSCALE>150</YSCALE>
<XPOS>20</XPOS>
<YPOS>100</YPOS>

Likewise, Card 2 is focused on Target B:
<XSCALE>220</XSCALE>
<YSCALE>220</YSCALE>
<XPOS>200</XPOS>
<YPOS>40</YPOS>

Assume that Cards 1 and 2 are arranged side-by-side in the program, and that the drawing sampling rate is 1/12 second. The Viewer then draws the visual material

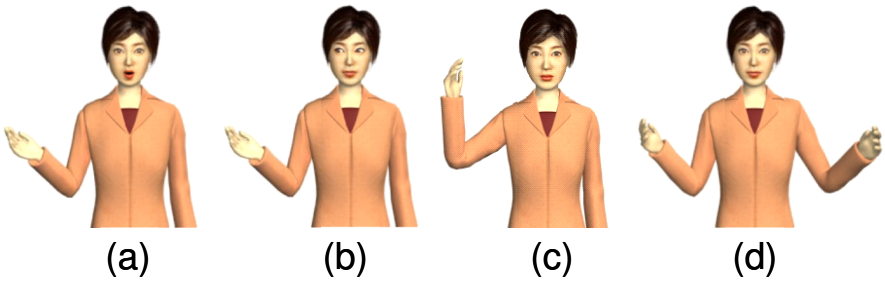


Figure 6.12: RISA snapshots. (a) RISA is doing a beat gesture, (b) RISA is looking away, (c) RISA is pointing at the visual material display, and (d) RISA is doing a "big" iconic gesture.

every 1/12 second while changing the X position of the material by (200-20)/12= 15 pixels and the Y position by (40-100)/12= -5 pixels. The same algorithm is applied to calculate the scale of the material, which is gradually changed from 150% to 220%. As a result, it appears in the Viewer as if a camera moves from Target A to Target B while zooming in on Target B. This technique can also be applied to a video clip.

The advantage of this method is that users need not design the camera work itself. They simply need to change the scale and position of a picture intuitively by manipulating a GUI. In TV programs, camera work (e.g., zoom, pan, tilt) is frequently used to improve the comprehensibility of a program. With the technique proposed here, such useful camera work is automatically generated in SPOC programs.

6.7.2 Playing Agent Animation

When all the animations and the synthesized voice are ready to play, the Viewer sends a cue to start playing them in a synchronized manner. To control the animated character through a web-based application, we implemented the RISA animation system in Macromedia Flash. Snapshots of RISA gesturing and changing her facial expression are shown in Figure 6.12.

The basic idea of this animation system is to construct an agent animation by assembling small animations of each body part. The agent body is divided into 12 parts: head, two eyebrows, two eyes, two eyeballs, mouth, two arms, and two hands. Small pieces of animations are defined for each body part (e.g., moving the left eyebrow up 30 degrees, moving the right arm in front of the body). The total number of actions in the library is over 300, including reverse actions for the hand gestures. By combining these animations, various kinds of agent behaviors can be produced.

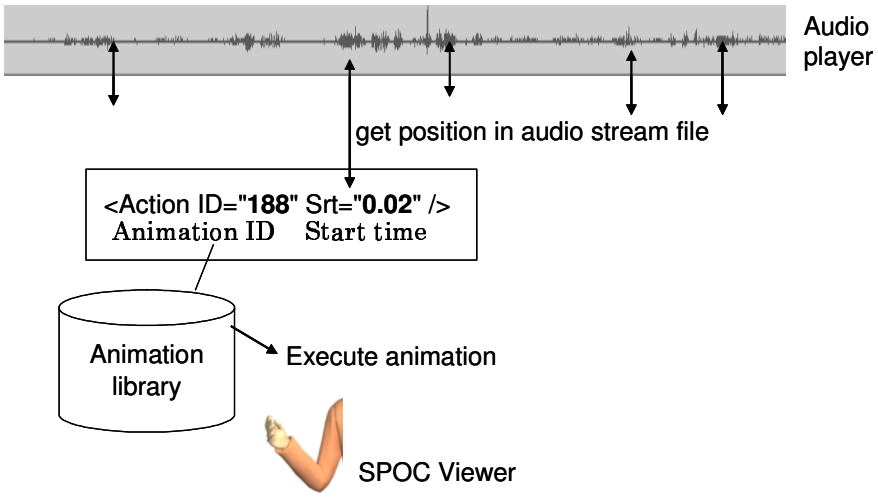


Figure 6.13: Synchronization between audio and animation

The SPOC Viewer generates agent animations by executing the agent action script specified in the <AGENT> element of a Knowledge Card XML. Figure 6.13 shows how the SPOC Viewer synchronizes the animations with speech. The agent speech is saved in the streaming server as an MP3 audio file, and it is played through an MP3 audio player. The Viewer accesses the MP3 player to get the current position in the audio file. If the current time matches the start time of an action, the Viewer picks up the animation for the action ID from the animation library and executes it. In addition to actions co-occurring with speech, RISA also performs some idle actions, such as looking away and blinking, during silent periods.

6.8 Evaluation

Our main concern in designing the SPOC system is to help users create multimedia story content without writing a script for a whole program. To accomplish this goal, we have proposed a very simple Knowledge Card editing interface. In this section, based on the results of two experiments, we examine whether SPOC is actually simple and easy enough for novice users to create multimedia content.

6.8.1 Experiment 1

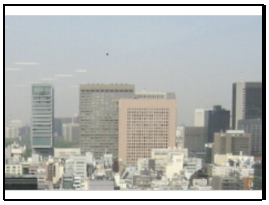
As a preliminary study, four subjects (two men and two women) learned how to edit a Card, and the times required for them to perform tasks were measured. In this experiment, the subjects were required to edit Cards correctly according to recipes for the Cards.

Card 2

<Graphics>
http://100.100.0.100:8500/warehouse/demo/105-0554_IMG.JPG

<Text>
私たちの街には、高層ビル、高層マンションが立ち並んでいます。
In our town, there are many tall buildings and apartments.

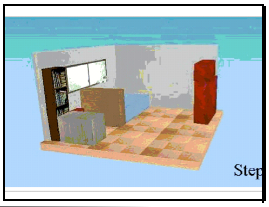
Card 3



<Graphics>
http://100.100.0.100:8500/warehouse/demo/105-0554_IMG.JPG
（zooming Card 2）

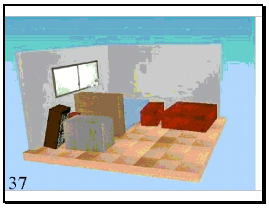
<Text>
では、地震が起こったとき、私たちの街はどうなる　のでしょうか。
Then, when an earthquake occurs, what happens in our town?

Card 5



＜Movie＞
http://100.100.0.100:8500/warehouse/demo/2.swf
（zooming left part of the movie ）
＜Text＞
こちらは、建物の一階部分の様子ですが、ほとんど被害は見られません。
The first floor is not shaking so much.

Card 6



＜Movie＞
http://100.100.0.100:8500/warehouse/demo/2.swf
（zooming right part of the movie）
＜Text＞
しかし、上層階では、家具が倒れるほどの揺れが起こります。
However, the top floor is shaking very strongly so that furniture falls down.

Figure 6.14: Example of Card recipes

Procedure: At the beginning of a session, an experimenter instructed the subjects on how to use the Knowledge Card Editor. After the instruction session, each subject performed a task by herself/himself.

Task: The subjects' task was to make a one-minute SPOC program about earthquake simulation with seven Knowledge Cards. For each Card, a recipe was provided to the subject. The recipe specified the name of a graphics/movie file to be selected, the text to be typed in, and a picture showing the zoom scale and the focused area. The average length of text in the Cards was 30 characters. The recipes for Cards 2 and 3 are shown in Figure 6.14. These two Cards used the same graphics, but the zoom scales and focused areas were different. Thus, digital camera work would be applied to these Cards. Likewise, camera work for a video clip would be applied to Cards 5 and 6.

Results: The mean length of the instruction session was 4:32, and the average time for the subjects to perform the task was 13:30. Thus, on average, a subject made a one-minute program in 13.5 minutes, after 4.5 minutes of instruction. Moreover, all of the subjects edited one Card in less than 2 minutes. These results suggest that learning the Card editing process was quite easy for all of the subjects. The operation time, however, does not indicate the users' impressions of the software. We therefore addressed this issue in Experiment 2.

6.8.2 Experiment 2

To examine users' subjective impressions of SPOC, we gathered another set of subjects (nine people: two women and seven men) and asked them to answer a questionnaire after creating an original SPOC program.

Procedure: At the beginning of a session, each subject was provided the same instruction as in Experiment 1. Each subject then edited a short practice program, which consisted of two Cards using the same graphical material and one Card using a video clip (i.e., Cards 2, 3, and 6 in Experiment 1).

Task: After practicing, the subjects were asked to create an original program about "Tokyo" with at least four Cards. As visual materials, 23 graphics files and 7 movie clips of Tokyo were provided. After creating their programs, the subjects answered a questionnaire asking about their impressions of watching the programs and using the SPOC system. The questions are listed in Table 6.2. The subjects answered these questions on a four-point scale.

Table 6.2: Questions about general impression on SPOC

(Q1)	The sequence of operation procedures was not complicated.
(Q2)	SPOC was easy to use.
(Q3)	It would not be long before I could use this software perfectly.
(Q4)	Creating a program with SPOC was enjoyable.
(Q5)	SPOC is useful.
(Q6)	SPOC can convey information comprehensibly.
(Q7)	SPOC can convey information accurately.
(Q8)	If I have the chance, I would like to use this software again.
(Q9)	If I familiarize myself with this software, I will be able to make more interesting programs than I made this time.

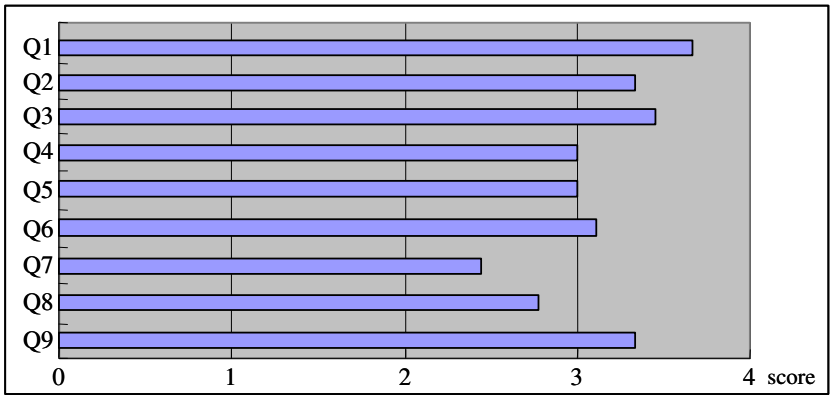


Figure 6.15: Results of subjective evaluation

Results: The mean length of the programs was 1 minutes, 13 seconds (SD = 10.8 sec) with 6.2 Cards. All of the subjects used video clips as well as graphics. On average, a video clip was used in 1.8 Cards per program. The users' general impressions of the software are shown in Figure 6.15. The subjects felt that the sequence of operations was not complicated (Q1, Q2), nor did it take time to learn (Q3). They also felt that SPOC was useful, and that it was comfortable and enjoyable to use (Q4, Q5). They also exhibited positive attitudes about using the software again (Q8, Q9). An interesting finding is that the subjects judged SPOC as conveying information more comprehensibly (Q6), rather than accurately (Q7). In addition to the general impressions, we also asked about usability of each operation. We found that some operations, such as file selection and time range specification for a video clip, still need to be improved.

Discussion: The results strongly support our approach. After receiving brief instructions, the subjects succeeded in not only editing Cards correctly according to recipes, but also in creating their own programs. In Experiment 2, all of the subjects created highly original, well-structured programs. The stories expressed the subjects' opinions about Tokyo, as well as their personal lives and experiences there. We did not, however, examine how well these programs would have been received by community members. Through further work, we need to evaluate SPOC programs from the recipient point of view and examine how they affect communication in a community.

6.9 Related work

6.9.1 Methods for multimedia contents generation

Multimedia content generation has been studied mainly from two different approaches: the scripting approach, and the AI approach. We thus compare these approaches with our method used in SPOC and CAST.

In the scripting approach, content developers create scripts for multimedia content by using markup languages, such as SMIL (SMIL, web page), MPML (Ishizuka et al., 2000), and TVML (Hayashi, Ueda, and Kurihara, 1999). SMIL is a markup language for describing synchronized multimedia in general. MPML defines a set of tags for controlling a Microsoft Agent. TVML is a markup language for creating a TV program with a background setting, character animations, and camera work. Although these markup languages provide specifications for designing multimedia content in detail, content creators need to be skillful and patient enough to describe every single piece of the content, including when and how camera work is applied, and when and what type of nonverbal behaviors an animated agent should perform.

The other method is the AI-based approach, in which multimedia content is generated using artificial intelligence techniques such as planning (André, Rist, and Muller, 1999; Rickel and Johnson, 1999; Shaw, Johnson, and Ganeshan, 1999). This approach has developed highly sophisticated methods for automatic presentation generation. However, it also requires content creators to be skilled at describing relationships between events and actions in logical form.

In contrast, the combination of SPOC and CAST enables users to create multimedia content without scripting all the details or learning logic and AI theory. SPOC automatically generates camera work by using the size and position information of the visual material in a Knowledge Card. CAST calculates an agent's action script from text by analyzing linguistic information. Using this environment, users can easily create multimedia content. As for agent animation generation, CAST was inspired by (Cassell, Vilhjalmsson, and Bickmore, 2001). In that case,

the researchers proposed a toolkit that takes plain text as input and automatically suggests a sequence of agent behaviors synchronized with synthesized speech. We have elaborated on this idea and proposed more detailed gesture generation rules specifically for Japanese.

In addition, SPOC employs Knowledge Cards as the building blocks of programs, while other markup languages provide GUI-based program editors for editing the whole sequence of a program (e.g., TVML Editor). We suggest that the Card interface makes it easier for non-expert users to construct programs from small pieces. On the other hand, a program editor allows more professional users to design programs in more detail. Thus, choosing an appropriate system according to the purpose of communication is important.

6.9.2 Web-based Presentation Agent

With the goal of providing a media technology that does not depend on either computer performance or platform, we have implemented our system as a web application. Because web-based applications have become so popular, a newscaster agent on a web TV has actually been implemented for commercial use (ANANOVA, web page). This system employs a scripting approach, so that all of the agent's behaviors are described by content designers. As more basic research, multimodal web-based presentation generation has been studied based on the AI approach. For example, in PPP persona (André, 1997), multimodal help instructions are generated and presented on the web by an animated agent. Adele, developed at USC, is a pedagogical agent working on a web-based medical education system (Shaw, Johnson, and Ganeshan, 1999). Note that these systems are designed to help users learn something by watching multimedia content. In contrast, SPOC-CAST aims to help users not only in viewing multimedia content but also in creating their own content.

6.10 Summary and discussion

This paper has described a web-based multimedia environment, SPOC. The system allows non-expert users to create multimedia story content, like a TV program, and to play programs posted by other community members. SPOC can use both video clips and graphics as visual materials, and automatically generates digital camera work from these materials. Moreover, it automatically determines and generates agent animations based on linguistic information in a text. Our evaluation experiments showed that SPOC is easy to learn and use for creating and playing programs. These results suggest that SPOC can contribute to reducing the volume of user tasks in creating multimedia content, while also encouraging users to spend

more time engaged in communication with other community members.

In Section 6.5, we have addressed the issues related to assigning gestures to text and converting the text into agent animations synchronized with speech. First, our empirical study identified useful lexical and syntactic information for assigning gestures to plain text. Specifically, when a bunsetsu unit is a constituent of coordination, gestures occur almost half the time. Gestures also frequently co-occur with nominal phrases modified by a clause. These findings suggest that syntactic structure is a stronger determinant of gesture occurrence than theme or rheme and given or new information specified by local grammatical cues.

We plan to enhance CAST by incorporating more general discourse level information, though the current system does exploits cue words as a very partial kind of discourse information. For instance, gestures frequently occur at episode boundaries. Pushing and popping of a discourse segment (Grosz and Sidner, 1986) may also affect gesture occurrence. Therefore, by integrating a discourse analyzer into the Language Tagging Module (LTM), more general discourse information can be used in the CAST mechanism.

Although the evaluation experiments focused on content creation, evaluating SPOC from the audience's perspective will be an important future work. For example, it is important to investigate whether a SPOC program can affect a viewer's attitude, and how viewers respond to a program that they have watched. Such bidirectional evaluation will be necessary to improve the communication functionality of this system.

In addition, it is also necessary to evaluate the effectiveness of nonverbal agent behaviors in actual human-agent interaction. We expect that if CAST can generate nonverbal behaviors with appropriate timing for emphasizing important words and phrases, users will perceive agent presentations as lively and comprehensible. An important future direction for our research will be conducting a user study to examine this hypothesis.

Chapter 7

Discussion and Future Direction

7.1 Overall discussion

Previous chapters have shown how to design and build MCIs based on a model of human communication derived from empirical studies. Chapters 3 and 4 proposed content planners, while Chapter 5 addressed dialogue state management. Chapter 6, finally, presented a multimodal generation module for agent animations and speech. These chapters mainly contribute to the design of the *Conceptualization Module* in MCI. There remain, however, certain issues as yet unaddressed in this thesis.

First, the thesis does not provide sufficient discussion concerning the understanding portion of MCI. Studies on multimodal understanding proposed methods for interpreting user's intentions by integrating inputs from multiple modalities, such as speech and pointing (Wahlster, 1991; Johnston and Bangalore, 2000; Bolt, 1980). In addition to such behavior directly contributing to interpretation of meaning, very subtle nonverbal signals would also be useful for reducing recognition errors. For example, users may not refer to an object that is unseen, and they may not look away from the system (conversational agent) when they ask a question. Therefore, user's gaze direction would be a useful indicator to check whether the system interpretation of user input is reasonable in a given communication situation.

Second, we admit that this thesis did not address sentence realization mechanisms in multimodal generation. The systems described in Chapters 3 and 4 use a sentence realization mechanism by Kato et al. (1996b). Simple template-based sentence generation is used in MACK, meanwhile, in Chapter 5. Although Chapter 6 addressed gesture generation, a part of the multimodal realization mechanism, it did not address the generation of linguistic expressions. Integrating more sophisticated multimodal realization mechanisms (Cassell, Stone, and Yan, 2000) would improve MCIs in allowing the generation of a variety of multimodal expressions, including gestures and linguistic expressions, according to the communication situation.

While Chapter 6 focused on gestures as paralinguistic information accompanying speech, prosody is also important as paralinguistic information, and is strongly linked to gestures, as pointed out by (Kendon, 1972). Consequently, it is expected that the gesture assigning mechanism proposed in Chapter 6 may be used for assigning intonational accents to a sentence, and presenting a richer multimodal output to users.

7.2 Evaluation scheme

Although evaluation is one of the essential phases in our research approach, we admit that the evaluation scheme for MCIs has not yet been established well. In this thesis, we measured the following aspects in evaluation experiments:

(1) Accuracy of the model: How accurately does the system predict given behaviors?

(2) Effectiveness of the system: Whether and how much does the system reduce the cost and burden of the user in using the system?

(3) Naturalness of interaction: How natural is the interaction between the agent and the user?

As an extension of (2), it would be important to measure systems effectiveness in user's achievement of a task through the communication with the system.

Evaluating user interfaces, specifically evaluating them with respect to effectiveness in interaction with users, is not so simple as measuring system performance. Therefore, establishing a well-structured evaluation scheme would be indispensable for evolving such a new research field. For this purpose, we need more discussion about what evaluation dimensions would be necessary, what milestones would be possible for each dimension, and how these dimensions interact with each other. We believe that establishing an evaluation scheme would contribute to making the research progress clear and figuring out what aspects need to be studied more.

7.3 Diversity and universality in communicative behaviors

In this research, empirical studies have been conducted for the purpose of defining general rules for communicative behaviors, which are independent of cultural and ethnic difference. We collected good amount of human behavior data and analyzed them statistically, and then established models and rules based on the statistical results. We admit that not only finding the universality of communicative behaviors,

but also describing their diversity is also indispensable. Behavior selection rules should be customized according to the cultural and ethnic difference. However, statistical analysis is mainly used in analyzing behaviors of people in western countries, and ethnographic methods are employed in studying minority people, such as African cultures. We hope that assimilating the findings from both approaches would contribute to modeling the universality and the diversity of communicative behaviors, and customizing behavior selection rules.

7.4 Future directions

7.4.1 Producing contents for multimodal communication

While the mechanisms proposed in this thesis attempt to produce improved human-computer interaction, the preparation of contents used in the interaction remains a big problem.

In multimodal dialogue systems similar to those presented in Chapters 3 and 4, the contents of the communication need to be prepared in advance by writing plan recipes and rules for selecting the utterance contents. As the content selection mechanisms are independent of the domain knowledge, changing the domain of conversation is theoretically possible. However, creating a new domain knowledge costs considerably high.

On the contrary, in the presentation system outlined in Chapter 6, contents are generated automatically from plain text, which can be collected very easily. These contents, however, are less interactive than the conversations exchanged in multimodal dialogue systems. Moreover, the presentation contents are not interlinked with each other, and cannot be dynamically re-constructed or automatically re-used.

I believe that one possible approach for improving the content creation mechanism is to build a multimodal conversational environment, where a set of multimodal story contents (e.g., presentations), each of which is automatically generated from plain text, are loosely connected with the background scene and with each other. In such an environment, the user can interact with a conversational agent by traversing the story-network, and visual information in the environment can provide a meaningful and consistent connection between the stories told by the agent. In addition, topic similarity between stories, which can be calculated using a natural language processing technique, would also represent a useful link to interconnect such story contents. All these linguistic and visual contexts would be integrated to dynamically construct a conversation between a user and an MCI.

7.4.2 Improving the reality of communication

While the primary concern of this thesis is to improve the naturalness of human-computer interaction using multiple communication modalities, the reality of communication would also be an important aspect. We think that the more natural nature of the communication would contribute to its reality.

In face-to-face communication, conversation is affected by what is existing and taking place within the environment, such as objects and walls in a room, or buildings and walking people outside. It is not realistic to imagine two people talking in an empty place. Thus, we think that one of the key issues for improving the reality of human-MCI communication is enhancing the MCIs' awareness of the situation, and the ability to recognize situational information and exploit the information to get the user involved in the conversation.

For example, MACK, a virtual character presented in Chapter 5, cannot directly manipulate the physical map. MACK can, however, share the information on the map by sensing the places that the user points to on the map. This mechanism enables MACK to be aware of the physical world in which the user lives. In addition, if a conversational agent can control the user's attention in the virtual world, the agent's capability in terms of attention control would contribute toward the reality of communication. For example, it would be possible for an agent to control the user's attention using nonverbal behaviors, such as gazing and pointing at the area referred to and focused on in the conversation.

One of the problems in enhancing agent awareness of background situation in a virtual conversational environment is to build a background itself. When we first developed the MID-3D system presented in Chapter 4, the graphics technology available was insufficient to construct a high quality virtual environment. Later on, thanks to considerable advances in computer graphics, 3D graphics have been becoming much more realistic, and used in conversational interfaces (Traum and Rickel, 2002; Traum et al., 2003). In these systems, background scenes are very realistic. However, creating such a realistic 3D background still requires enormous effort and only professional CG designers are able to do it.

In our new project, we are developing a conversational environment using a panoramic picture as a background, creating a background far more easily than in a 3D environment. As shown in Figure 7.1, conversational agents are shown in the background, and talk about objects and events also shown in the background. In Figure 7.1, the agents are at the scene of an earthquake disaster and discuss what is happening in the background. By adding some annotation to the background, the agents become able to be aware of their background. In this system, the contents creators can easily develop a new situation to which conversational agents have access when presenting a story.

Figure 7.1: Embodied conversational agent embedded in a background

Moreover, in such an immersive conversational environment, human users may regards conversational agents as conversational partners living in the virtual world, and on the basis of the shred environment, the users expect to establish a longer term relationship with the agents. To support such a relationship, conversational agents need to be adaptive to the communication style of the users as well as their personal profiles. We expect that machine learning techniques are useful for this issue.

7.4.3 Other communication artifacts

While this thesis has mainly focused on conversational virtual agents, there would be other kinds of communicative artifacts. Communication robots living in the same physical world alongside humans are becoming increasingly popular. Robots live in the physical world, and can change the world by manipulating objects. On the contrary, conversational agents live in a virtual world, and cannot change the physical world. Although these two types of artifacts work in differing environments, both can exploit bodily expressions, and demonstrate communicative capability using verbal and nonverbal behaviors. While research on communication robots has just started, and these robots can perform only limited types of communicative behaviors, such as moving the head towards the user (Sidner et al., 2004; Miyauchi et al., 2004), we believe that models and algorithms for conversational agents may be applied to the design and construction of communication robots. Specifically, adaptivity in regulating a communication with human user would be an important

common issue between communication robots and conversational agents.

We believe that these communicative artifacts will allow human users more natural and intuitive communication with computer systems, and will become one of the most popular human interfaces in the near future.

Chapter 8

Conclusion

Aiming to design computer artifact able to communicate with a human user in a natural way, this thesis covers issues relating to multimodal conversational interfaces. In Chapter 1, we proposed a basic architecture of a multimodal conversational interface (MCI). Chapter 2, meanwhile, reviews related studies on linguistics, computational linguistics, human communication, and human interface. Individual studies presented in the rest of the chapters contribute to some modules in the MCI architecture. Chapters 3 and 4 presented content planners and discussed the means of deciding multimodal utterance contents according to the state of dialogue and characteristics of the topic. Chapter 5 addresses face-to-face grounding through nonverbal behaviors, and implements our face-to-face grounding model into an embodied conversational agent. Chapter 6, finally, was concerned with multimodal generation, especially that of agent gestures and integrated an agent behavior generation module into an automatic presentation generation system.

This thesis has addressed a wide range of issues concerning the design of multimodal conversational interfaces, and presented prototype systems demonstrating how proposed methods work. More importantly, the system designs proposed here are supported by empirical studies, involving the collection and analysis of considerable amounts of real human communication data through enormous time and effort. We have illuminated aspects of multimodal communication by bridging between empirical methods of human behavior science and media technologies to build conversational interfaces. This thesis has been accomplished by employing the spiral approach; consisting of the following steps:

1. Empirical Study

- Data collection
- Annotation
- Statistical analysis

2. Establishing a model

3. System implementation

4. Evaluation.

As multimodal conversational interfaces are becoming more popular and used in practical situations, such as education and network communities, it will become more important to take into account both how people interact with others and how these communication skills are revealed in communication with a computer artifact. Specifically in terms of the evaluation phase, studies in communication science (Reeves and Nass, 1996) may provide useful ideas for human-computer interaction experiments.

Although, with the use of current technologies, interaction with a computer artifact is still a long way from genuine face-to-face communication, I believe that this thesis has contributed to this issue, and demonstrated a new direction of research on human-computer interaction.

Appendix A

Instruction of experiment in Chapter 5

<Instruction for a direction giver drawing a map>
"First, here is a piece of paper and a pen. I would like you to draw a map from X to Y to Z on this piece of paper. I will give you extra pieces so that you can re-draw the map if you don't like it. But, don't worry. You don't need to draw a perfectly accurate or beautiful map. A rough sketch is fine. The only requirement is to draw at least 8 landmarks or signs in the map. Please do not draw only lines. Draw a map from X to Y to Z, OK? Any questions?"

<Instruction for direction giving task in F2F condition> "Hi... (whatever greetings). Okay, here is the task: I'd like you to give (direction giver's name) directions from X to Z by passing through each of the landmarks on the map. You are welcome to use the map that you drew earlier and use the pen to add more details if you need to. You can take as much time as you need, just make sure that (direction receiver's name) gets to each landmark before you go on to the next leg of the directions (address the receiver). You have to really understand how to get to each landmark before (direction giver's name) goes on to the next step of the directions. So, when you (address the receiver) really understand how to get to a landmark, you move your piece to there. Any questions?"

References

Allen, James and Mark Core. 1997. Draft of DMSL: Dialogue act markup in several layers, http://www.cs.rochester.edu/research/cisd/resources/damsl/revisedmanual/revisedmanual.html.

Allen, James, George Ferguson, and Amanda Stent. 2001. An architecture for more realistic conversational systems. In *Intelligent User Interfaces 2001 (IUI-01)*.

Allen, James F., Bradford W. Miller, Eric K. Ringger, and Teresa Sikorski. 1996. A robust system for natural spoken dialogue. In *the 1996 Annual Meeting of the Association for Computational Linguistics (ACL'96)*, pages 62–70.

Allen, James F. and C. Raymond Perrault. 1980. Analyzing intention in utterances. *Artificial Intelligence*, 15(3):143–178.

Allwood, Jens, Joakim Nivre, and Elisabeth Ahlisen. 1992. On the semantics and pragmatics of linguistic feedback. *Journal of Semantics*, 9:1–26.

ANANOVA. web page. http://www.ananova.com.

Anderson, A.H., E. Bard, C. Sotillo, G. Doherty-Sneddon, and A. Newlands. 1997. The effects of face-to-face communication on the intelligibility of speechs. *Perception and Psychophysics*, 59:580–592.

Anderson, Ann H., Miles Bader, Ellen Gurman Bard, Elizabeth Boyle, Gwyneth Dohetry, Simon Garrod, Stephen Isard, Jacqueline Kowtko, Jan McAllister, Jim Miller, Catherine Sotillo, Henry Thompson, and Regina Weinert. 1991. The hcrc map task corpus. *Language and Speech*, 34(4):351–366.

André, Elisabeth. 1997. Animated interface agents, making them intelligent. In *15th International Joint Conference on Artificial Intelligence (IJCAI-97)*.

André, Elisabeth. 2000. The generation of multimedia documents. In R. Dale, H. Moisl, and H. Somers, editors, *A Handbook of Natural Language Processing: Techniques and Applications for the Processing of Language as Text*. Marcel Dekker Inc., pages 305–327.

André, Elisabeth, T. Rist, and J. Muller. 1999. Employing ai methods to control the behavior of animated interface agents. *Applied Artificial Intelligence*, 13:415–448.

André, Elisabeth and Thomas Rist. 1993. The design of illustrated documents as a planning task. In Mark T. Maybury, editor, *Intelligent Multimedia Interfaces*. AAAI Press / The MIT Press, pages 94–116.

Andre, Elisabeth, Thomas Rist, Susanne van Mulken, Martin Klesen, and Stephan Baldes. 2000. The automated design of believable dialogues for animated presentation teams. In Justine Cassell, Joseph Sullivan, Scott Prevost, and Elizabeth Churchill, editors, *Embodied Conversational Agents*. MIT Press, Cambridge, MA, pages 220–255.

Argyle, M. and J. Graham. 1977. The central europe experiment - looking at persons and looking at things. *Journal of Environmental Psychology and Nonverbal Behaviour*, 1:6–16.

Argyle, Michael and Mark Cook. 1976. *Gaze and Mutual Gaze*. Cambridge University Press, Cambridge.

Argyle, Michael, Roger Ingham, Florisse Alkema, and Margaret McCallin. 1973. The different functions of gaze. *Semiotica*, VIII(1):19–32.

Austin, John Langshaw. 1962. *How to Do Things with Words*. Harvard University Press.

Aylett, R. 1999. Narrative in virtual environments - towards emergent narrative. In *Narrative Intelligence, AAAI Fall Symposium 1999*, volume Technical Report FS-99-01, pages 83–86. AAAI Press.

Bares, William H. and James C. Lester. 1997. Realtime generation of customized 3D animated explanations for knowledge-based learning environments. In *Proceedings of the 14th National Conference on Artificial Intelligence*, pages 347–354.

Bavelas, Janet Beavin, Nicole Chovil, Linda Coates, and Lori Roe. 1995. Gestures specialized for dialogue. *Personality and Social Psychology Bulletin*, 21:394–405.

Berger, K. W. and G. R. Popelka. 1971. Extra-facial gestures in relation to speechreading. *Journal of Communication Disorders*, 3:302–308.

Bolt, Richard A. 1980. Put-that-there: voice and gesture at the graphics interface. *Computer Graphics*, 14(3):262–270.

Boyle, E., A. Anderson, and A. Newlands. 1994. The effects of visibility in a cooperative problem solving task. *Language and Speech*, 37(1):1–20.

Brennan, Susan. 2000. Processes that shape conversation and their implications for computational linguistics. In *38th Annual Meeting of the ACL*.

Bruce, Bertram C. 1975. Generation as a social action. In *Theoretical Issues in Natural Language Processing-1*. pages 64–67.

Bruner, Jerome S. 1990. *Acts of Meaning*. Harvard University Press, Boston.

Bull, P. E. 1987. *Posture and Gesture*. Pergamon Press.

Carberry, Sandra. 1990. *Plan Recognition in Natural Language Dialogue Acts of Meaning*. The MIT Press, Cambridge.

Carletta, Jean. 1992. *Risk-taking and Recovery in Task-Oriented Dialogue*. Ph.D. thesis, University of Edinburgh.

Cassell, J., T. Bickmore, L. Campbell, H. Vilhjalmsson, and H. Yan. 2001. More than just a pretty face: Conversational protocols and the affordances of embodiment. *Knowledge-Based Systems*, 14:55–64.

Cassell, Justine. 2000. Nudge, nudge, wink, wink: Elements of face-to-face conversation for embodied conversational agents. In Justine Cassell, Joseph Sullivan, Scott Prevost, and Elizabeth Churchill, editors, *Embodied Conversational Agents*. MIT Press, pages 1–27.

Cassell, Justine, Tim Bickmore, Lee Campbell, Hannes Vilhjalmsson, and Hao Yan. 2000. Human conversation as a system framework: Designing embodied conversational agents. In Justine Cassell, Joseph Sullivan, Scott Prevost, and Elizabeth Churchill, editors, *Embodied Conversational Agents*. MIT Press, pages 29–63.

Cassell, Justine and Scott Prevost. 1996. Distribution of semantic features across speech and gesture by humans and computers. In L.S. Messing, editor, *Workshop on the Integration of Gesture in Language and Speech*, pages 253–270, Newark, DE. WIGLS.

Cassell, Justine, Matthew Stone, and Hao Yan. 2000. Coordination and context-dependence in the generation of embodied conversation. In *INLG 2000*, pages 171–178, Mitzpe Ramon, Israel. Association of Computational Linguistics.

Cassell, Justine and Kristinn R. Thorisson. 1999. The power of a nod and a glance: Envelope vs. emotional feedback in animated conversational agents. *Applied Artificial Intelligence*, 13:519–538.

Cassell, Justine, Hannes Vilhjalmsson, and Timothy Bickmore. 2001. Beat: The behavior expression animation toolkit. In *SIGGRAPH 01*, pages 477–486, Los Angeles, CA. ACM Computer Graphics Press.

Cawsey, Alison. 1990. A computational model of explanatory discourse; local interactions in a plan-based explanation. In Luff Paul, Nigel Gilbert, and Dai Frohlich, editors, *Computers and Conversation*. Academic Press.

Cawsey, Alison. 1992. *Explanation and Interaction: The Computer Generation of Expalanatory Dialogues*. The MIT Press.

Clark, Herber H. and Deanna Wilkes-Gibbs. 1986. Referring as a collaborative process. *Cognition*, 22:1–39.

Clark, Herbert H. 1994. Managing problems in speaking. *SPEECH COMMUNICATION*, 15(3-4):243–250.

Clark, Herbert H. 1996. *Using language*. Cambridge University Press.

Clark, Herbert H. 2003. Pointing and placing. In S. Kita, editor, *Pointing: Where language, culture, and cognition meet*. Hillsdale NJ: Erlbaum.

Clark, Herbert H. and Susan E. Brennan. 1991. Grounding in communication. In L. B. Resnick, J. M. Levine, and S. D. Teasley, editors, *Perspectives on socially shared cognition*. American Psychological Association, pages 127–149.

Clark, Herbert H. and Edward F. Schaefer. 1989. Contributing to discourse. *Cognitive Science*, 13:259–294.

Cohen, Phillip R. 1984. The pragmatics of referring, and the modality of communication. *Computational Linguistics*, 10:97–146.

Cohen, Phillip R. and C. Raymond Perrault. 1979. Elements of a plan-based theory of speech acts. *Cognitive Science*, 3(3):177–212.

Core, Mark and Jame Allen. 1997. Coding dialogue with the damsl annotation scheme. In *AAAI Fall Symposium on Communicative Action in Humans and Machines*.

Craig, S. D. and B. Gholson. 2002. Does an agent matter? : The effects of animated pedagogical agents on multimedia environments. In Barker P. and Rebelsky S., editors, *ED-MEDIA 2002: World Conference on Educational Multimedia, Hypermedia and Telecommunications*, pages 357–362.

Dahal, M., S. Feiner, K. McKeown, S. Pan, M. Zhou, T. H'ollerer, J. Shaw, Y. Feng, and J. Fromer. 1996. Negotiation for automated generation of temporal multimedia presentations. In *ACM Multimedia96*, pages 55–64. ACM Press.

Daly-Jones, O., A. Monk, and L. Watts. 1998. Some advantages of video conferencing over high-quality audio conferencing: fluency and awareness of attentional focus. *International Journal of Human-Computer Studies*, 49(1):21–58.

Dillenbourg, Pierre, David Traum, and Daniel Schneider. 1996. Grounding in multimodal task-oriented collaboration. In *EuroAI&Education Conference*.

Duncan, Starkey. 1972. Some signals and rules for taking speaking turns in conversations. *Journal of Personality and Social Psychology*, 23(2):283–292.

Duncan, Starkey. 1974. On the structure of speaker-auditor interaction during speaking turns. *Language in Society*, 3:161–180.

Ekman, P. and W. V. Friesen. 1969. The repertoire of nonverbal behavior: Categories, origins, usage, and coding. *Semiotica*, 1(1):49–98.

Exline, Ralph V. and B.J. Fehr. 1982. The assessment of gaze and mutual gaze. In Klaus R. Scherer and Paul Ekman, editors, *Handbook of methods in nonverbal behavior research*. Cambridge University Press, Cambridge, pages 91–135.

Feiner, Steven K. and Kathleen R. McKeown. 1993. Automating the generation of coordinated multimedia presentations. In Mark T. Maybury, editor, *Intelligent Multimedia Interfaces*. AAAI Press / The MIT Press, pages 117–138.

Ferguson, George and James F. Allen. 1998. TRIPS: An integrated intelligent problem-solving assistant. In *Proceedings of the 15th National Conference on Artificial Intelligence*, pages 567–572.

Firbas, J. 1971. On the concept of communicative dynamism in the theory of functional sentence perspective. *Philologica Pragensia*, 8:135–144.

Fukuhara, T., N. Fujihara, S. Azechi, H. Kubota, and T Nishida. 2003. A network-based interactive broadcasting system for supporting a knowledge-creating community. In R.J.Howlett, N.S.Ichalkaranje, L.C.Jain, and G.Tonfoni, editors, *Internet-Based Intelligent Information Processing Systems*. World Scientific Publishing, pages 227–268.

Garau, M., M. Slater, S. Bee, and M.A. Sasse. 2001. The impact of eye gaze on communication using humanoid avatars. In *SIG-CHI conference on Human factors in computing systems*, pages 309–316.

Givon, T. 1985. Iconicity, isomorphism and non-arbitrary coding in syntax. In J. Haiman, editor, *Iconicity in Syntax*. John Benjamins, pages 187–219.

Goodwin, Charles. 1981. *Conversational Organization: Interaction between speakers and hearers*. Academic Press, New York.

Grice, H. P. 1975. Logic and conversation. In P. Cole and J. L. Morgan, editors, *Syntax and Semantics. Vol. 3: Speech Acts*. Academic Press.

Grosz, Barbara J. and Candace L. Sidner. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Hadar, Uri. 1989. Two types of gesture and their role in speech production. *Language and Social Psychology*, 8:221–228.

Halliday, M. A. K. 1967. *Intonation and Grammar in British English.* Mouton, The Hague.

Hayashi, M., H. Ueda, and T. Kurihara. 1999. TVML (TV program making language) - automatic TV program generation from text-based script -. In *Imagina'99*, pages 31–42.

Heeman, Peter A. and Graeme Hirst. 1995. Collaborating on referring expressions. *Computational Linguistics*, 21(3):163–200.

Hobbs, Jerry R. 1979. Coherence and coreference. *Cognitive Science*, 3(1):67–90.

Horiuchi, Yasuo, Yukiko Nakano, Hanae Koiso, Masato Ishizaki, Hiroyuki Suzuki, Michio Okada, Makiko Naka, Syun Tutiya, and Akira Ichikawa. 1999. The design and statistical characterization of the japanese map task dialogue corpus (in japanese). *Transactions of the Japanese Society for Artificial Intelligence*, 14(2):63–74.

Hovy, E. H. 1993. Automated discourse generation using discourse structure relations. *Artificial Intelligence*, 63:341–385.

Hovy, Eduard. 1975. Approaches to the planning of coherent text. In Cécile L. Paris, William R. Swartout, and William C. Mann, editors, *Natural Language Generation in Artificial Intelligence and Computational Linguistics.* Kluwer, Boston, pages 83–102.

Isaacs, W. N. 1996. The process and potential of dialogue in social change. *Educational Technology*, Jan.Feb.:20–30.

Ishikawa, Yukiko and Tsuneaki Kato. 1991. Conversation between operator and customer: An analysis of operator's inquiry (in japanese). In *Sixth Annual Meeting of JSAI.*

Ishizuka, M., T. Tsutsui, S. Saeyor, H. Dohi, Y. Zong, and H. Prendinger. 2000. Mpml: A multimodal presentation markup language with character agent control functions. In *Agents2000 Workshop 7 on Achieving Human-like Behavior in Interactive Animated Agents*, pages 51–54, Barcelona, Spain.

Johnson, W. L., J. W. Rickel, and J. C. Lester. 2000a. Animated pedagogical agents: Face-to-face interaction in interactive learning environments. *International Journal of Artificial Intelligence in Education*, 11:47–78.

Johnson, W. Lewis, Jeff W. Rickel, and James C. Lester. 2000b. Animated pedagogical agents: Face-to-face interaction in interactive learning environments. *International Journal of Artificial Intelligence in Education*.

Johnston, Michael and S. Bangalore. 2000. Finite-state multimodal parsing and understanding. In *COLING-2000*, Saarbruecken, Germany.

Kapoor, Ashish and Rosalind W. Picard. 2001. A real-time head nod and shake detector. In *Workshop on Perceptive User Interfaces*.

Kato, Tsuneaki, Yukiko I. Nakano, Hideharu Nakajima, and Takaaki Hasegawa. 1996a. Interactive multimodal explanations and their temporal coordination. In *ECAI-96*, pages 261–265. John Willey and Sons Limited.

Kato, Tsuneaki, Yukiko I. Nakano, Hideyaru Nakajima, and Takaaki Hasegawa. 1996b. Interactive multimodal explanations and their temporal coordination. In *the 12th European Conference on Artificial Intelligence (ECAI-96)*, pages 261–265. John Wiley and Sons Limited.

Kawanobe, Akihisa, Susumu Kakuta, Hirofumi Touhei, and Katsumi Hosoya. 1998. Preliminary report on HyCLASS authoring tool. In *ED-MEDIA/ED-TELECOM*.

Kendon, Adam. 1967. Some functions of gaze direction in social interaction. *Acta Psychologica*, 26:1–47.

Kendon, Adam. 1972. Some relationships between body motion and speech. In A W Siegman and B Pope, editors, *Studies in Dyadic Communication*. Pergamon Press, Elmsford, NY, pages 177–210.

Kerpedjiev, S., G. Carenini, S. F. Roth, and J. D. Moore. 1997. Integrating planning and task-based design for multimedia presentation. In *the 1997 International Conference on Intelligent User Interfaces*, pages 145–152.

Kiyota, Yoji, Sadao Kurohashi, and Fuyuko Kido. 2002. Dialog navigator : A questions answering system based on large text knowledge base. In *The 19th International Conference on Computational Linguistics (COLING 2002)*, pages 460–466, Taipei.

Kraft, R. 1986. The role of cutting in the evaluation and etention of film. *Journal of Experimental Psychology : Learning, Memory & Cognition*, 12(1):155– 163.

Kraut, R., M. Miller, and J. Siegel. 1996. Communication in performance of physical tasks: effects on outcomes and performance. In *the Conference on Computer Supported Cooperative Work*, pages 57–66. New York: ACM Press.

Kubota, H., S. Kurohashi, and T. Nishida. 2002. Virtualized-egos using knowledge cards. In *Seventh Pacific Rim International Conference on Artificial Intelligence(PRICAI-02) WS-5 International Workshop on Intelligent Media Technology for Communicative Reality (IMTCR2002)*, pages 51–54, Tokyo, JAPAN.

Kurohashi, Sadao and Makoto Nagao. 1994. A syntactic analysis method of long japanese sentences based on the detection of conjunctive structures. *Computational Linguistics*, 20(4):507–534.

Larsson, Staffan, Peter Bohlin, Johan Bos, and David Traum. 1999. Trindikit 1.0 (manual). Technical Report Trindi Project Deliverable D2.2.

Lawrence, D. and J. B. Thomas. 1999. Social dynamics of storytelling: Implications for story-base design. In *Narrative Intelligence, AAAI Fall Symposium 1999*, volume Technical Report FS-99-01. AAAI Press.

Lesh, Neal, Charles Rich, and Candace Sidner. 1999. Using plan recognition in human-computer collaboration. In *Conference on User Modeling*, pages 23–32. Springer Wien New York.

Lester, James C., Brian A. Stone, and Gray D. Stelling. 1999. Lifelike pedagogical agents for mixed-initiative problem solving in constructivist learning environments. *User Modeling and User-Adapted Interaction*, 9(1-2):1–44.

Lester, James C., Stuart Towns, Charles Callaway, Jennifer Voerman, and Patrick FitzGerald. 2000. Deictic and emotive communication in animated pedagogical agents. In Justine Cassell, Joseph Sullivan, Scott Prevost, and Elizabeth Churchill, editors, *Embodied Conversational Agents*. MIT Press, pages 123–154.

Lester, James C., Jennifer Voerman, Stuart Towns, and Charles Callaway. 1999. Deictic believability: Coordinating gesture, locomotion,and speech in lifelike pedagogical agents. *Applied Artificial Intelligence*, 13(4-5):383–414.

Lester, James C., Jennifer L. Voerman, Stuart G. Towns, and Charles B. Callaway. 1997. Cosmo: A life-like animated pedagogical agent with deictic believability. In *IJCAI-97 Workshop, Animated Interface Agent*.

Levelt, Willem J. M. 1989. *Speaking: From Intention to Articulation*. The MIT Press.

Levinson, Stephen C. 1983. *Pragmatics*. Cambridge University Press.

Linden, K. U. 1994. Generating precondition expressions in instructional text. In *Proceedings of the 32th Annual Meeting of the Association for Computational Linguistics*, pages 42–49.

Linden, K. U. and J. H. Martin. 1995. Expressing rhetorical relations in instructional text: A case study of the purpose relation. *Computational Linguistics*, 21(1):29–57.

Littman, Diane J. and James F. Allen. 1987. A plan recognition model for subdialogues in conversations. *Cognitive Science*, 11(2):163–200.

Lochbaum, Karen E. 1998. A collaborative planning model of intentional structure. *Computational Linguistics*, 24(4):525–572.

Mann, William C. and Sandra A. Thompson. 1987. Rhetorical structure theory: A theory of text organization. Technical Report ISI/RS-87-190, USC/ISI.

Mateas, M. and A. Stern. 2000. Towards integrating plot and character for interactive drama. In *Social Intelligent Agents: The Human in the Loop Symposium. AAAI Fall Symposium Series*. AAAI Press.

Matheson, Colin, Massimo Poesio, and David Traum. 2000. Modelling grounding and discourse obligations using update rules. In *1st Annual Meeting of the North American Association for Computational Linguistics (NAACL2000)*.

Maybury, Mark T. 1993. Planning multimedia explanation using communicative acts. In Mark T. Maybury, editor, *Intelligent Multimedia Interfaces*. AAAI Press / The MIT Press, pages 59–74.

McKeown, Kathleen R. 1985. *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Lnaugage Text*. Cambridge University Press.

McNeill, David. 1992. *Hand and Mind: What Gestures Reveal about Thought*. The University of Chicago Press, Chicago, IL/London, UK.

Mittal, V. O. and C. L. Paris. 1993. Generating natural language descriptions with examples: Differences between introductory and advanced texts. In *Proceedings of the 11th National Conference on Artificial Intelligence*, pages 271–276.

Miyauchi, Dai, Arihiro Sakurai, Akio Nakamura, and Yoshinori Kuno. 2004. Active eye contact for human-robot communication. In *CHI2004 Late breaking results paper*, pages 1099–1102, Vienna, Austria.

Mooney, D. J., S. Carberry, and K. F. McCoy. 1991. Capturing high-level structure of naturally occurring extended explanations using bottom-up strategies. *Computational Intelligence*, 7:334–356.

Moore, Johanna D. 1995. *Participating in Explanatory Dialogues: Interpreting and Responding to Questions in Context*. MIT Press.

Moore, Johanna D. and C. L. Paris. 1989. Planning text for advisory dialogues. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*.

Moore, Johanna D. and W. R. Swartout. 1989. A reactive approach to explanation. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, pages 1504–1510.

Morency, L.P., A. Rahimi, and T. Darrell. 2003. A view-based appearance model for 6 dof tracking. In *IEEE conference on Computer Vision and Pattern Recognition*.

Moser, Megan and Johanna D. Moore. 1996. Toward a synthesis of two accounts of discourse structure. *Computational Linguistics*, 22(3):409–420.

Mott, B. W., C. B. Challaway, and L. S. Zettlemoyer. 1999. Towards narrative-centered learning environments. In *Narrative Intelligence, AAAI Fall Symposium 1999*, volume Technical Report FS-99-01. AAAI Press.

Nakajima, Shin'ya and James F. Allen. 1992. Prosody as a cue for discourse structuree. In *ICSLP*, pages 425–428.

Nakanishi, Hideyuki, Chikara Yoshida, Toshikazu Nishimura, and Toru Ishida. 1996. Free walk: Supporting casual meetings in a network. In *ACM CSCW'96*, pages 308–314.

Nakatani, Christine and David Traum. 1999. Coding discourse structure in dialogue (version 1.0). Technical Report UMIACS-TR-99-03, University of Maryland.

Nass, Clifford, Katherine Isbister, and Eun-Ju Lee. 2000. Truth is beauty: Researching embodied conversational agents. In Justine Cassell, Joseph Sullivan, Scott Prevost, and Elizabeth Churchill, editors, *Embodied Conversational Agents*. MIT Press, pages 374–402.

Nishida, Toyoaki. 2002. Social intelligence design for web intelligence. *IEEE Computer:Special Issue on Web Intelligence*, Vol. 35(No. 11):37–41.

Noma, Tsukasa and Norman Badler. 1997. A virtual human presenter. In *IJCAI 97*.

Novick, David G., Brian Hansen, and Karen Ward. 1996. Coordinating turn-taking with gaze. In *ICSLP-96*, Proceedings of ICSLP-96, pages 1888–91, Philadelphia, PA.

Olson, J.S., G.M. Olson, and D.K. Meader. 1995. What mix of video and audio is useful for remote real-time work? In *CHI '95*, pages 362–368. ACM Press.

Paek, Tim and Eric Horvitz. 1999. Uncertainty, utility, and misunderstanding: A decision-theoretic perspective on grounding in conversational systems. In Susan E. Brennan, Alain Giboin, and David Traum, editors, *Working Papers of the AAAI Fall Symposium on Psychological Models of Communication in Collaborative Systems*, pages 85–92. American Association for Artificial Intelligence.

Paris, Cécile L. 1991. The role of the user's domain knowledge in generation. *Computational Intelligence*, 7(2):71–93.

Perrault, C. Raymond and James F. Allen. 1980. A plan-base analysis of indirect speech acts. *American Journal of Computational Linguistics*, 6(3-4):167–182.

Pierrehumbert, J. B. 1980. *The phonology and phonetics of english intonation.* Ph.D. thesis, Massachusetts Institute of Technology.

Prevost, Scott Allan. 1996. An informational structural approach to spoken language generation. In *34th Annual Meeting of the Association for Computational Linguistics*, Santa Cruz, CA.

Reeves, Byron and Clifford Nass. 1996. *The Media Equation: how people treat computers, televisions and new media like real people and places.* Cambridge University Press.

Rich, Charles and Candace L. Sidner. 1998. COLLAGEN: A collaboration manager for software interface agents. *User Modeling and User-Adapted Interaction*, 8:315–350.

Rickel, Jeff and W. Lewis Johnson. 1999. Animated agents for procedural training in virtual reality: Perception, cognition and motor control. *Applied Artificial Intelligence*, 13(4-5):343–382.

Rogers, W. 1978. The contribution of kinesic illustrators towards the comprehension of verbal behavior within utterances. *Human Communication Research*, 5:54–62.

Rosenfeld, Howard M. and Margaret Hancks. 1980. The nonverbal context of verbal listener responses. In Mary Ritchie Key, editor, *The Relationship of Verbal and Nonverbal Communication.* Mouton Publishers, New York, pages 194–206.

Sacks, H., E. A. Schegloff, and G. Jefferson. 1974. A simplest systematics for the organization of turn-taking in conversation. *Language*, 50(4):696–735.

Schank, R. and R Abelson. 1995. Knowledge and memory: the real story. In W Wyer, editor, *Advances in Social Cognition, vol. VII*. Lawrence Elbaum, Hillsdale, NJ, pages 1–86.

Schank, Roger C. and Robert P. Abelson. 1977. *Scripts, Plans, Goals and Understanding*. Lawrence Erlbaum, Hillsdale, N.J.

Searle, John R. 1975. Indirect speech acts. In P. Cole and J. Morgan, editors, *Syntax and Semantics. Vol.3: Speech Acts*. NewYork: Academic Press, pages 59–82.

Shaw, E., W.L. Johnson, and R. Ganeshan. 1999. Pedagogical agents on the web. In *the Third International Conference on Autonomous Agents*, pages 283–290.

Sidner, Candace. 1994. An artificial discourse language for collaborative negotiation. In *12th Intnl. Conf. on Artificial Intelligence (AAAI)*, pages 814–819. MIT Press.

Sidner, C.L., C.D. Kidd, C.H. Lee, and N.B. Lesh. 2004. Where to look: A study of human-robot engagement. In *ACM International Conference on Intelligent User Interfaces (IUI)*, pages 78–84.

Sinclair, John McH. and Malcolm Coulthard. 1990. *Plan Recognition in Natural Language Dialogue Acts of Meaning*. The MIT Press, Cambridge.

SMIL. web page. The synchronized multimedia integration language (SMIL), http://www.w3.org/audiovideo/.

Stenström, Anna-Brita. 1994. *An Introduction to Spoken Interaction*. Longman.

Stent, Amanda, John Dowding, Jean Mark Gawron, Elizabeth Owen Brat, and Robert Moore. 1999. The CommandTalk spoken dialogue system. In *ACL99*, pages 183–190.

Stone, Brian A. and James C. Lester. 1996. Dynamically sequencing an animated pedagogical agent. In *Proceedings of the 13th National Conference on Artificial Intelligence*, pages 424–431.

Takahashi, T. and H: Takeda. 2001. Telmea: An asynchronous community system with avatar-like agents. In Michitaka Hirose, editor, *The Eighth IFIP TC.13 Conference on Human-Computer Interaction (INTERACT 2001)*, pages 190–197.

Towns, Stuart G., Charles B. Callaway, and James C. Lester. 1998. Generating coordinated natural language and 3D animations for complex spatial explanations. In *Proceedings of the 15th National Conference on Artificial Intelligence*, pages 112–119.

Traum, David and Peter Heeman. 1996. Utterance units and grounding in spoken dialogue. In *ICSLP*.

Traum, David and Jeff Rickel. 2002. Embodied agents for multi-party dialogue in immersive virtual worlds. In *the first International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2002)*, pages 766–773.

Traum, David, Jeff Rickel, Jonathan Gratch, and Stacy Marsella. 2003. Negotiation over tasks in hybrid human-agent teams for simulation-based training. In *the Second International Joint Conference on Autonomous Agents and Multi-Agent Systems(AAMAS 2003)*, pages 441–448.

Traum, David R. 1994. *A Computational Theory of Grounding in Natural Language Conversation*. Ph.D. thesis, University of Rochester.

Traum, David R. and Pierre Dillenbourg. 1998. Towards a normative model of grounding in collaboration. In *ESSLLI-98 workshop on Mutual Knowledge, Common Ground and Public Information*, pages 23–32. Springer Wien New York.

van Mulken, S., E. Andre, and J. Mfiller. 1998. The persona effect: How substantial is it? In H. Johnson, L. Nigay, and C. Roast, editors, *People and Computers XIII: Proceedings of HCI'98*, pages 53–66.

Vertegaal, Roel and Yaping Ding. 2002. Explaining effects of eye gaze on mediated group conversations: Amount or synchronization? In *CSCW 2002 Conference on Computer Supported Collaborative Work*. ACM Press.

Wahlster, W., N. Reithinger, and A. Blocher. 2001. Smartkom: Multimodal communication with a life-like character. In *Eurospeech 2001, 7th European Conference on Speech Communication and Technology*, pages 1547–1550.

Wahlster, Wolfgang. 1991. User and discourse models for multimodal communication. In Joseph W. Sulivan and Sherman W. Tyler, editors, *Intelligent User Interfaces*. acm press, pages 45–67.

Wahlster, Wolfgang, Elisabeth André, Wolfgang Finkler, Hans-Jürgen Profitlich, and Thomas Rist. 1993. Plan-based integration of natural language and graphics generation. *Artificial Intelligence*, 63:387–427.

Walker, M. A. 1992. Redundancy in collaborative dialogue. In *Proceedings of the 14th International Conference on Computational Linguistics*, pages 345–351.

Walker, M. A. and O. Rambow. 1994. The role of cognitive modeling in achieving communicative intentions. In *Proceedings of the 7th International Workshop on Natural Language Generation*.

Whittaker, S. 2003. Theories and methods in mediated communication. In A. Graesser, editor, *The Handbook of Discourse Processes*. MIT Press.

Whittaker, S. and B. O'Conaill. 1993. Evaluating videoconferencing. In *Companion Proceedings of CHI'93 Human Factors in Computing Systems*. ACM Press.

Publicaion List

Journal Papers

1. Yukiko I. Nakano, Toshihiro Murayama, Masashi Okamoto, Daisuke Kawahara, Qing Li, Sadao Kurohashi, and Toyoaki Nishida. Cards-to-presentation on the web: Generating multimedia contents featuring agent animations. *Journal of Network and Computer Applications*, (to appear).

2. Yukiko I. Nakano, Toshihiro Murayama, and Toyoaki Nishida. Multimodal story-based communication: Integrating a movie and a conversational agent. *IEICE Transactions on Information and Systems*, Vol. E87-D, No. 6, pp. 1338–1346, 2004.

3. Yasuo Horiuchi, Yukiko Nakano, Hanae Koiso, Masato Ishizaki, Hiroyuki Suzuki, Michio Okada, Makiko Naka, Syun Tutiya, and Akira Ichikawa. The design and statistical characterization of the japanese map task dialogue corpus. *Journal of Japanese Society for Artificial Intelligence*, 14(2):261–272, 1999. (in Japanese)

4. Tsuneaki Kato and Yukiko I. Nakano. Generation of interactive multimodal explanation -two frameworks of explanation and interactivity-. *Journal of Japanese Society for Artificial Intelligence*, 14(3):455–465, 1999. (in Japanese)

5. Yukiko I. Nakano and Tsuneaki Kato. Instruction generation using dialogue history and user model. *Transactions of the Information Processing Society of Japan*, 38(11):2179–2190, 1997. (in Japanese)

6. Tsuneaki Kato and Yukiko I. Nakano. Referring actions in multi-modal dialogues. *Journal of Japanese Society for Artificial Intelligence*, 12(4):627–634, 1997. (in Japanese)

7. Yukiko Ishikawa (Nakano) and Tsuneaki Kato. Utterance content planner for generating conversational query expression - deciding utterance content according to characteristics of query content -. *Journal of Japanese Society for Artificial Intelligence*, 10(6):962–970, 1995. (in Japanese)

8. Yukiko Ishikawa (Nakano) and Takashi Muto. Context-dependency in making requests - social role as a situational determinant of requesting forms -. *Japanese Journal of Educational Psychology*, 38. (in Japanese)

International Conferences

1. Yukiko I Nakano, Masashi Okamoto, Daisuke Kawahara, Quing Li, and Toyoaki Nishida. Converting text into agent animations: Assigning gestures to text. In *Proceedings of Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2004), Companion Volume*, pp. 153–156, 2004.

2. Yukiko I. Nakano, Toshihiro Murayama, and Toyoaki Nishida. Engagement in situated communication by conversational agents. In *Multimedia and Network Information Systems, vol. 2: the 1st International Workshop on Intelligent Media Technology for Communicative Intelligence*, pp. 95–101. Affiliated with 4th Polish National Conference on Multimedia and Network Information Systems, (2004).

3. Yukiko I. Nakano, Masashi Okamoto, and Toyoaki Nishida. Enriching agent animations with gestures and highlighting effects. In *Proceedings of International Workshop on Intelligent Media Technology for Communicative Intelligence*, pp. 112–115, 2004.

4. Ken'ichi Matsumura, Yukiko I. Nakano, and Toyoaki Nishida. The analysis of conversational contents creation process on spoc system. In *Proceedings of International Workshop on Intelligent Media Technology for Communicative Intelligence*, pp. 100–103, 2004.

5. Masashi Okamoto and Yukiko I. Nakano. Toward enhancing user involvement via empathy channel in human-computer interface design. In *Proceedings of International Workshop on Intelligent Media Technology for Communicative Intelligence*, pp. 129–132, 2004.

6. Qing Li, Yukiko Nakano, Masashi Okamoto, and Toyoaki Nishida. Highlighting multimodal synchronization for embodied conversational agents. In *Proceedings of the 2nd International Conference on Information Technology for Application (ICITA 2004)*, 2004.

7. Yukiko I. Nakano, Gabe Reinstein, Tom Stocky, and Justine Cassell. Towards a model of face-to-face grounding. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL 03)*, pp. 553–561, 2003.

8. Yukiko I. Nakano, Toshihiro Murayama, Daisuke Kawahara, Sadao Kurohashi, and Toyoaki Nishida. Embodied conversational agents for presenting intellectual multimedia contents. In *Proceedings of the 7th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES 2003)*, pp. 1030–1036, 2003.

9. Toshihiro Murayama, Yukiko Nakano, and Toyoaki Nishida. Participatory broadcasting systemusing interface agent and multimedia. In *Proceedings of Social Intelligence Design International Conference (SID 2003)*, 2003.

10. Yukiko I. Nakano. Nonverbal signals for natural communication with embodied conversational agents. In *Proceedings of International Workshop on Intelligent Media Technology for Communicative Reality*, 2002.

11. Justine Cassell, Tom Stocky, Timothy Bickmore, Yang Gao, Yukiko Nakano, Kimiko Ryokai, Dona Tversky, Catherine Vaucelle, and Hannes Vilhjálmsson. Mack: Media lab autonomous conversational kiosk. In *Proceedings of Imagina02*, 2002.

12. Justine Cassell, Yukiko I. Nakano, Timothy W. Bickmore, Candace L. Sidner, and Charles Rich. Non-verbal cues for discourse structure. In *Proceedings of the 39th Annual Meeting of the Association of Computational Linguistics (ACL 01)*, pp. 106–115, 2001.

13. Justine Cassell, Yukiko I. Nakano, Timothy W. Bickmore, Candace L. Sidner, and Charles Rich. Annotating and generating posture from discourse structure in embodied conversational agent. In *Proceedings of ACM Agents Conference Workshop on Representing, Annotating, and Evaluating Non-Verbal and Verbal Communicative Acts to Achieve Contextual Embodied Agents*, 2001.

14. Yukiko I. Nakano, Kenji Imamura, and Hisashi Ohra. Taking account of the user's view in 3d multimodal instruction dialogue. In *Proceedings of International Conference on Computational Linguistics (COLING 2000)*, pp. 572–578, 2000.

15. Yukiko I. Nakano and Tsuneaki Kato. Cue phrase selection in instruction dialogue using machine learning. In *Proceedings of COLING-ACLf98 Workshop for Discourse Relations and Discourse Markers*, pp. 100–106, 1998.

16. Tsuneaki Kato and Yukiko I. Nakano. Aggregative utterance planning for interactive instruction. In *Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP-97)*, pp. 120 – 126, 1997.

17. Takaaki Hasegawa, Yukiko I. Nakano, and Tsuneaki Kato. A collaborative dialog model based on interaction between reactivity and deliberation. In Lewis Johnson, editor, In *Proceedings of the First International Conference on Autonomous Agents*, pp. 75–82. ACM SIGART, ACM Press, 1997.

18. Tsuneaki Kato and Yukiko I. Nakano. Towards generation of fluent referring action in multimodal situations. In *Proceedings of a Workshop on Referring Phenomena in a Multimedia Context and Their Computational Treatment*, pp. 20–28. ACL SIG on Multimedia, 1997.

19. Tsuneaki Kato, Yukiko I. Nakano, Hideyaru Nakajima, and Takaaki Hasegawa. Interactive multimodal explanations and their temporal coordination. In *Proceedings of the 12th European Conference on Artificial Intelligence (ECAI-96)*, pp. 261–265. John Wiley and Sons Limited, 1996.

20. Yukiko I. Nakano and Tsuneaki Kato. Task context dependency of explanation strategy in instruction dialogues. In *Proceedings of AAAI-95 Fall Symposium Series; Embodied Language & Action*, pp. 94–100. The American Association for Artificial Intelligence, 1995.

21. Tsuneaki Kato and Yukiko I. Nakano. Referent identification requests in multimodal dialogues. In *Proceedings of the International Conference on Cooperative Multimodal Communication (CMC/95)*, pp. 175–191. the Universities of Brabant Joint Research Organization and ACL SIG on Multimedia, Part II, 1995.

22. Yukiko Ishikawa. Communicative mode dependent contribution from the recipient in information providing dialogue. In *Proceedings of International Conference on Spoken Language Processing (ICSLP)*, pp. 959–962. the Universities of Brabant Joint Research Organization and ACL SIG on Multimedia, Part II, 1994.

23. Yukiko Ishikawa and Tsuneaki Kato. Deciding appropriate query content according to topic features. In *Proceedings of the first conference of the Pacific Association for Computational Linguistics (PACLING)*, pp. 232–241, 1993.

24. Tsuneaki Kato and Yukiko Ishikawa. Generating appropriate queries for dialogue. In *Proceedings of the 2nd Pacific Rim International Conference of Artificial Intelligence (PRICAI92), vol.2*, pp. 1190–1196. Korea Information Science Society, 1992.

25. Tsuneaki Kato and Yukiko Ishikawa. Ellipsis in japanese discourse. In *Proceedings of the Natural Language Processing Pacific Rim Symposium (NLPRS)*, pp. 217–223. Information Processing Society in Japan, 1991.

Domestic conferences

1. Yukiko Nakano, Masashi Okamoto, and Quing Li. Assigning gestures based on linguistic information - generating gestures for presentation agents -. In *the 10th Annual Meeting of the Association for Natural Language Processing (NLP04)*, pages 552–555, 2004. (in Japanese)

2. Yukiko Nakano, Toshihiro Murayama, and Toyoaki Nishida. Participatory broadcasting system featured with multimedia contents and interface agents. In *Forum on Information Technology 2003 (FIT 2003)*, pages 427–428, 2003. (in Japanese)

3. Yukiko I. Nakano. Dialogue management and multimodal generation for instruction dialogue systems. In *the 24th Annual Meeting for the Japanese Society for Information and Systems in Education (JSiSE99)*, pages 79–82, 1999. (in Japanese)

4. Yukiko I. Nakano and Kenji Imamura. Instruction dialogue system in 3D virtual environment. In *the 13th Annual Conference of the Japanese Society for Artificial Intelligence*, pages 64–67, 1999. (in Japanese)

5. Yukiko Nakano and Kenji Imamura. Integrating collaborative dialogue system with 3D virtual environment. In *Technical Report of the Institute of Electronics, Information and Communication Engineers, ET99-17*, pages 57–64, 1999. (in Japanese)

6. Yukiko I. Nakano, Makiko Nakano, Yasuo Horiuchi, Aya Yoshino, Syun Tutiya, Akira Ichikawa, Masato Ishizaki, Michio Okada, Hanae Koiso, and Hiroyuki Suzuki. Basic statistics of the japanese map task corpus (1). In *Technical Report of the Japanese Society for Artificial Intelligence, SIG-SLUD-9701-4*, pages 19–24, 1997. (in Japanese)

7. Yukiko I. Nakano and Tsuneaki Kato. Correcting user's misunderstanding in instruction dialogue: Generating repair dialogue using dialogue history. In *the 10th Annual Conference of the Japanese Society for Artificial Intelligence, 13-12*, pages 363–366, 1996. (in Japanese)

8. Yukiko I. Nakano Tsuneaki Kato. Utterance content and dialogue strategies in instruction dialogue: Effectsofthe discourse history and the understanding level of the novice. In *Technical Report of the Japanese Society for Artificial Intelligence, SIG-SLUD-9502-4*, pages 24–31, 1995. (in Japanese)

9. Yukiko I. Nakano and Tsuneaki Kato. A strategy for repeated explanation in instruction dialogues. In *the 50th Conference of the Information Processing Society of Japan, 4R-6, Vol. 3*, pages 95–96, 1995. (in Japanese)

10. Yukiko Ishikawa. Interaction between information giver and information receiver. In *the 48th Conference of the Information Processing Society of Japan, 7R-6, Vol. 3*, pages 229–230, 1994. (in Japanese)

11. Aono Motoko, Akira Ichikawa, Hanae Koiso, Shinji Satoh Makiko Naka, Syun Tutiya, Kenji Yagi, Naoya Watanabe, Masato Ishizaki, Michio Okada, Hiroyuki Suzuki, Yukiko Nakano, and Keiko Nonaka. The japanesse map task corpus: An interim report. In *Technical Report of the Japanese Society for Artificial Intelligence, SIG-SLUD-9402-5*, 1994. (in Japanese)

12. Yukiko Ishikawa and Tsuneaki Kato. Topic-dependency of inquiry: Selecting felicious question expressions by using topic features. In *Technical Report of the Information Processing Society of JapanC92-NL-88-4*, pages 25–32, 1992. (in Japanese)

13. Yukiko Ishikawa and Tsuneaki Kato. Conversation between operator and customer: Analysis of operators' inquiry. In *the 5th Annual Conference of the Japanese Society for Artificial Intelligence, 13-7*, pages 547–550, 1991. (in Japanese)

Other publications

1. Yukiko Nakano. Media technology for knowledge circulation: Employing interface agents. *Shakai Gijyutu Ronbunshuu*, 1. (in Japanese)

2. Yukiko Nakano, Toshihiro Murayama, and Toyoaki Nishida. Providing information through conversational agents: Emphasizing important concepts using nonverbal information. *Shakai Gijyutu Ronbunshuu*, 2. (in Japanese)

3. Akimichi Omura, Misako Ogino, Toshihiko Endo, Etuko Haryu, Yukiko Ishikawa, and Izumi Shirasa. Mother-child interaction in picture-book reading situation (the 1st report): Categories of mothers' utterances. *Research-aid Paper of the Yasuda Life Welfare Foundation*, 25(2):24–33, 1989. (in Japanese)