A Multimodal Galaxy-based Geographic System

by

Sy Bor Wang

B.S., Carnegie Mellon University, Pittsburgh, Pennsylvania(2001)

Submitted to the Department of Electrical Engineering
and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2003

Author .....
Department of Electrical Engineering and Computer Science
May 23rd, 2003

Certified by .....
Timothy James Hazen
Research Scientist
Thesis Supervisor

Certified by .....
Scott Cyphers
Research Scientist
Thesis Supervisor

Accepted by .....
Arthur Smith
Chairman, Departmental Committee on Graduate Students

A Multimodal Galaxy-based Geographic System

by

Sy Bor Wang

Submitted to the Department of Electrical Engineering
and Computer Science
on May 23rd, 2003, in partial fulfillment of the
requirements for the degree of
Master of Science

Abstract

With recent advances in technology, software applications are faced with the challenge of creating next generation human computer interfaces that offer users greater expressive power, naturalness, and portability. These systems will combine natural input modes such as speech, pen, hand gestures, eye gaze, and head and body movement or orientation, to create a meaningful multimodal input. They are also known as "multimodal" systems. The design of multimodal systems depends heavily on the knowledge of the natural integration patterns that represent people's combined use of different input modes.

The Galaxy Architecture, developed by the Spoken Language Systems Group, provides a framework for turn based conversational systems[14]. It has enabled the creation of applications in a variety of domains. One of these doamins is Voyager, which provides navigation assistance and traffic status information for Boston via a web interface. Users interact with Voyager using a spoken interface, which is less efficient than an interface that also provides pen and speech input. Introducing pointing and sketching gestures to the system can make verbalizing spatial instructions simpler, even eliminating the need to use speech in some instances. These gestures will make Voyager a more natural and powerful navigation interface. This thesis adds pen-based pointing and sketching gestures to Voyager, and adds complementary features to the system which were not previously possible without the gesture modalities. In the process, a multimodal framework for integration was designed. User experiments were later conducted to analyze how well gestures can boost the speech recognition scores of deictic words.

Thesis Supervisor: Timothy James Hazen
Title: Research Scientist

Thesis Supervisor: Scott Cyphers
Title: Research Scientist

Acknowledgments

I would like to thank my supervisors T.J. Hazen and Scott Cyphers for their kind and patient guidance in helping me accomplish my thesis. There have been many times when I committed silly mistakes or was slow at getting things done, but they were very patient and encouraging while teaching me the skills necessary to complete it.

I would also like to thank my advisor Jim Glass, for his kind support and funding of the project.

My lab mates were really wonderful. If not for their help, I am not sure how much I could have done for this thesis. Ed Filisko was very helpful and readily available at all times whenever I needed help on the context resolution. Min Tang taught me useful advice in C programming and Karen Livescu was readily available for any generic help that I had to get my thesis running. Jonathan Lau also helped me out in Java programs.

And to my parents, my two elder sisters for their undying support to my work in MIT.

This research was supported by an industrial consortium supporting the MIT Oxygen Alliance.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With recent advances in technology, software applications are faced with the challenge of creating next generation human computer interfaces that offer users greater expressive power, naturalness, and portability. These systems will combine natural input modes such as speech, pen, hand gestures, eye gaze, and head and body movement or orientation, to create a meaningful multimodal input. They are also known as "multimodal" systems.

During human communication, it is often the case that some ideas are more effectively conveyed in one mode than in others. Spatial information is more easily communicated by pointing at objects than by verbal description only, while relational information is more easily described by words than by pointing. Different modalities can also be used in conjunction with each other. During conversations, people often nod their heads to express agreement while saying something affirmative simultaneously. To express refusal, people can say something negative while their hands wave vehemently. Since these natural forms of interaction involve different modes of input, multimodal interfaces or systems that enable natural human computer interaction have been studied and different types of interfaces have emerged.

1.1 Background

Ever since Bolt's "Put That There"[1] demonstration system, which processed speech and manual pointing simultaneously, a variety of multimodal systems have emerged. Some systems recognize speech while determining the location of pointing from a user's manual gestures or gaze[8]. The Quickset system[2] integrates speech with pen input that includes drawn graphics, symbols, gestures, and pointing. Johnston recently developed MATCH(Multimodal Access To City Help)[7], a mobile multimodal speech-pen interface to restaurant and subway information for New York City. All these systems show that the modalities of speech and other gestures have different strengths and weaknesses, but combine to create a synergy where each modality complements the other. In the process, the users are provided with greater expressive power, naturalness and flexibility.

The design of multimodal systems depends heavily on the knowledge of the natural integration patterns that represent people's combined use of different input modes. The Quickset system uses a semantic unification process to combine the meaningful multimodal information carried by two input signals, while MATCH uses finite state methods for the multimodal integration process. According to Oviatt, the current challenge in multimodal systems is determining the integration and synchronization requirements for combining dynamically different modes within systems.[10]

1.2 Outline

The Galaxy Architecture developed by the Spoken Language Systems Group provides a framework for turn based conversational systems[14]. It has enabled the creation of applications in a variety of domains, such as a restaurant guide[5] and weather information[15]. The system used within this thesis is Voyager, which provides navigation assistance and traffic status information for Boston.

Voyager can respond to requests such as, "Show me driving directions from MIT to Harvard," or, "Show me a map of Boston." A map with the desired result will be

displayed on a Web interface. These queries can be spoken or typed into a text box. Interacting with these maps using only text or speech has some limitations. Temporal spatial information needs to be conveyed in a detailed lengthy manner. Complexities involving the expression of driving directions between unknown regions on the map require long descriptions. For example, a user may need to say "Show me the driving directions from the Boston downtown area close to Boston Commons to the end of Massachusetts Avenue that is south of Harvard." If pointing and sketching gestures are handled by the system, then verbalizing these instructions could be simpler, even eliminating the need for speech in some instances. For example, the user could point to a location on the map and say, "Show me the restaurants over here," or circle two regions on the map and say, "Show me the driving directions from here to here." The introduction of these gestures will make Voyager a more natural and powerful navigation interface.

As Voyager was implemented with a spoken interface using the Galaxy System, the goal of this thesis is to integrate and synchronize pen gestures with processed speech within Voyager. To achieve this, a multimodal framework for integration was designed. This framework was implemented with several new servers within the GALAXY architecture. These included a "multimodal" database, a new module within the context resolution server for integrating streams from multiple modalities, and a new graphical user interface that enables pen inputs.

Chapter 2

Voyager - a navigation interface

2.1 System Design of Voyager

An ideal navigation interface should be capable of displaying maps, locating landmarks, providing driving directions and providing real time traffic reports. The pre-existing Voyager system fulfills these capabilities. It has a map display, a text box for data entry, and a few buttons for control. The system engages in a turn-based dialog with the user, who either speaks to it directly or types questions into the text box. Although this system has had considerable success in serving the user's needs in navigation, the user's interaction with the system is mainly verbal. Other input modalities are mostly unused. For example, spatial selection and graffiti could not be processed by the system.

In human to human interactions, we speak and gesture to each other simultaneously, and information from our gestures and speech may complement each other. With Voyager, such gestures could be expressed as pen strokes or graffiti on the map. With such a wealth of new possibilities, we decided to introduce pen gestures into Voyager. Since pen gestures are a new input modality for Voyager that may be used with or without speech input, the design of Voyager needed to be re-structured. In the process, some new components were added and others were modified.

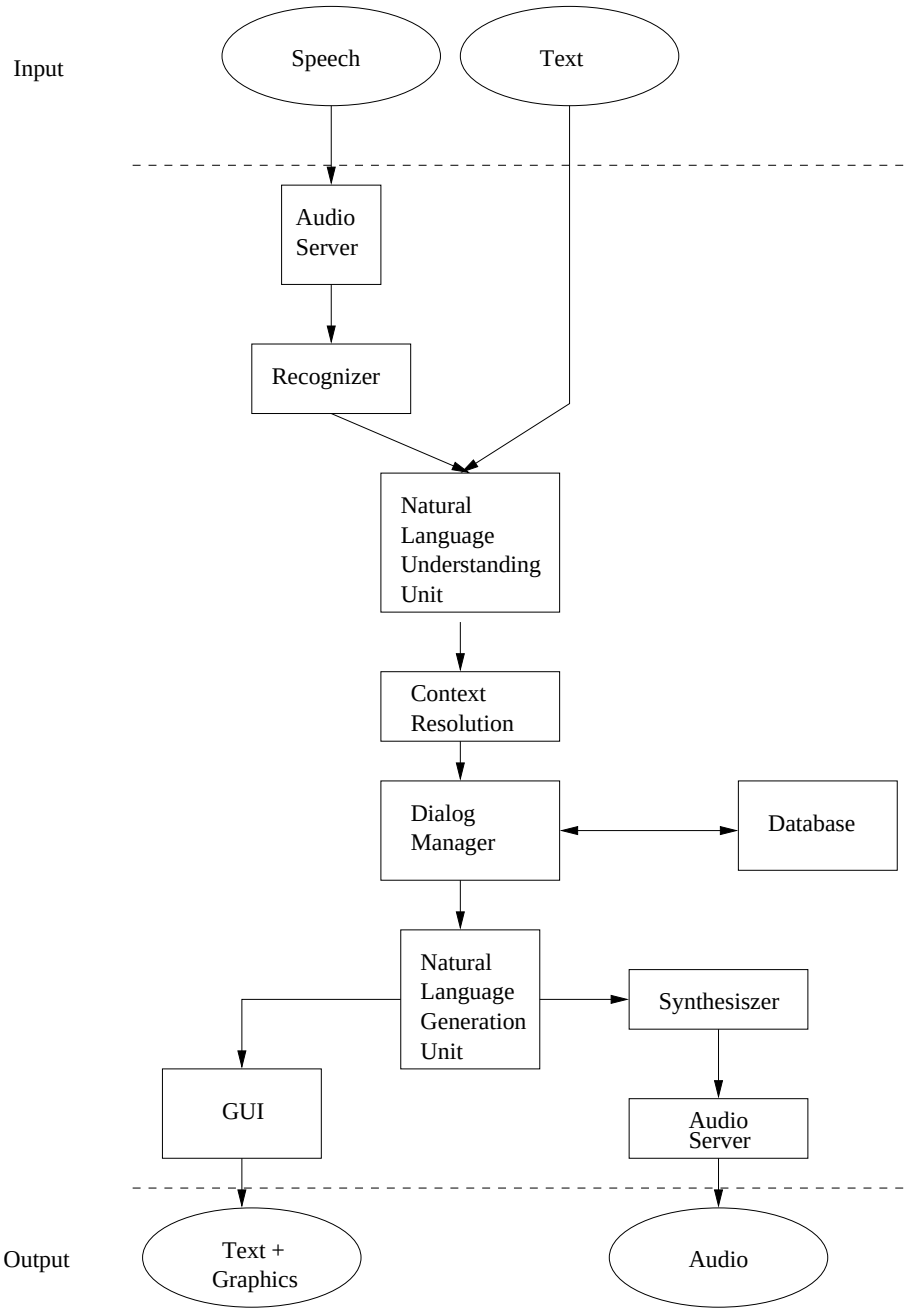

Figure 2-1: Components of the existing GALAXY System processing only speech and text

2.1.1 Processing Using Speech and Text Input

Speech and text inputs are processed by the existing Galaxy framework. Speech input is digitized and sent to the SUMMIT Speech recognizer[4] to process the audio waveform and generate a word graph of hypotheses. The word graph is parsed by TINA[13], the natural language understanding system, which selects and parses the best hypothesis and encodes it as a semantic frame. The semantic frame is then processed by the context resolution server to resolve discourse phenomena such as pronominal references, anaphoras and ellipsis[3]. The resolved semantic frame is then sent to the dialog manager, which determines the appropriate reply. This reply is sent to the natural language generation unit, which creates the text output of the reply and hands it to the synthesizer. Using this text output, the synthesizer puts together an audio representation of it and sends the audio waveform to the audio server.

Text input is processed in a similar manner, except that it is passed directly to the natural language understanding unit bypassing the speech recognizer. On the output side, the text output bypasses the synthesizer and audio server and is displayed on a graphical interface. Figure 2-1 shows the flow diagram of this system.

2.1.2 Pen Gesture Input

Unlike text and speech, pen gestures are expressed visually and may incorporate spatial information. Different gestures convey different meanings. For example, a circle implies focus over the enclosed region while a line implies focus over the region near it. A gesture recognizer was implemented to recognize such gestures. A semantic representation for the gestures was defined, and the context resolution process augmented to incorporate the additional information, so that speech, gesture and text could be combined in a consistent manner to disambiguate the user's intentions.

2.1.3 Synchrony of the input modalities

The combination of information from speech and gestures requires that both modalities are synchronized with each other. The word graphs of hypotheses generated by

the recognizer contain the start and end times of each word. Pen gestures were logged in a database, which we call the "multimodal database." The times the gestures occurred and their implied meaning are stored. This database is a new addition to Voyager. The information from the gestures is used by the context resolution server for fusion of gestures with speech information.

2.1.4 Fusion mechanism

The existing context resolution module resolved ambiguous references spoken by the user by incorporating information from the user's previous history of utterances. In a pen-based multimodal system, the implied meaning of pen gestures need to be incorporated into the semantic frame of the utterance as well. For example, if the user says, "How do I get there from MIT?", the system needs to determine if the user is referring to a region he circled on the map, or to a location he had spoken about in the previous utterance. The fusion mechanism to resolve this issue needed to be implemented.

2.1.5 GUI

Apart from the input processes, the display of Voyager also needed to be re-designed as well. The new interface accepts pen gestures from the user and provides a visual feedback about what was drawn on the map.

2.2 New Components of the system

In view of these design considerations, a new framework of Voyager was proposed. This new Voyager System has the following components:

- Graphical User Interface(GUI) - The new GUI, which can handle pen based input. All stroke gestures are drawn on this interface. It also contains a text entry box, a list box for listing results from database, and a message window

displaying the reply from the dialog manager. In operating systems without a pen, the mouse acts as a substitute.

- Gesture Recognizer - the gesture recognizer interprets simple gestures scribbled on the interface. At this time, only circles, lines and points are recognized. Since these gestures are rather easily distinguishable, a heuristic method is employed to recognize these simple strokes.

- Multimodal database - this database serves as a mouse event logger, recording all the mouse movements and their implied meaning during the user's interaction with the user.

- Updated Context Resolution server - this server will resolve references using both the implied meaning of the pen gestures and the references in the history of the dialog.

A flow diagram of the system is shown in Figure 2-2 The next few chapters will cover the design of the new components and the modified ones, namely, the graphical user interface(Chapter 3), the gesture recognizer(Chapter 4), the multimodal database(Chapter 5) and the multimodal context resolution mechanism(Chapter 6).

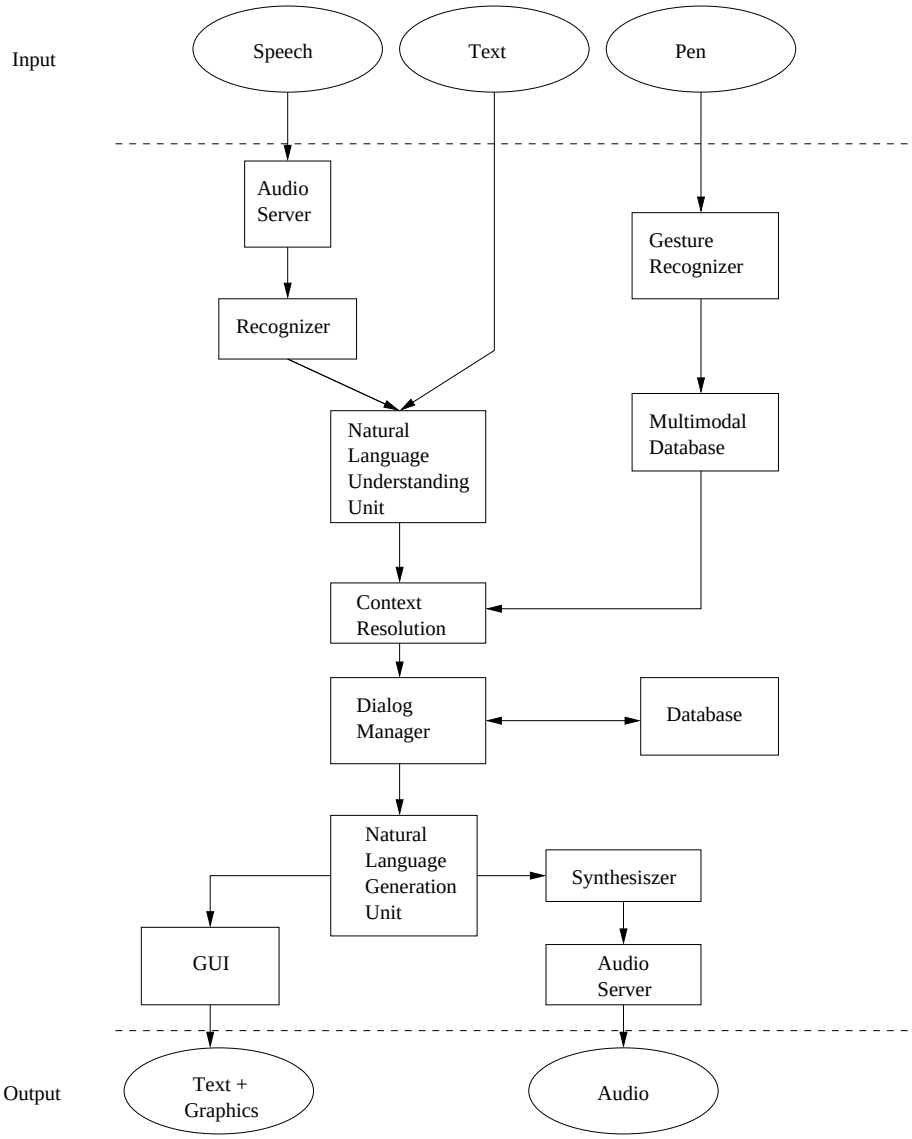

Figure 2-2: Components of the New System with the introduction of pen gestures

Chapter 3

The Graphical User Interface(GUI)

3.1 Components of the GUI

Figure 3-1 shows an image of the new graphical user interface. This interface contains a map display, a keyboard entry box, a message box and a list box listing results from Voyager.

- **Map Display:** This display shows an image containing the requested map. If the user asks for landmarks on the map, icons representing these landmarks are displayed as squares. When the mouse is over the display, it switches from an arrow icon to a small white point, like the tip of a pen, so that the user will think of the mouse pointer as a pen for drawing on the map. Any mouse clicks or mouse drags on the map will be represented as white points or thin white strokes respectively, which serves as visual feedback to the user.

- **List Box:** The list box shows database results or labels representing the icons on the map. When the user moves the mouse over the icon of interest, the icon and the list item are highlighted. This helps the user know the correct list item representing the icons.

- **Message Window:** The message window displays the verbal response of the computer. The replies from the computer are displayed separately from the

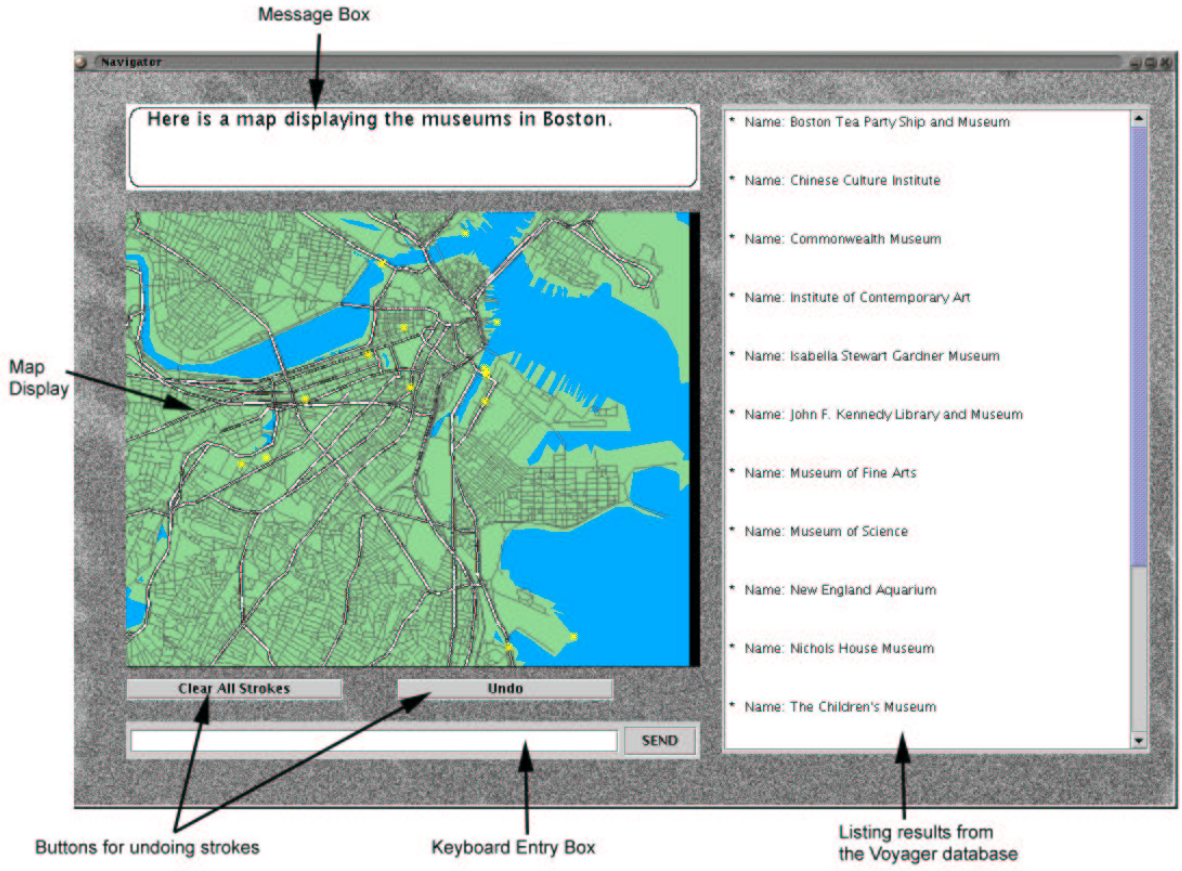

Figure 3-1: The Graphical User Interface.

database results because multiple turns could occur which utilize the same list of database results.

- **Keyboard Entry Box:** This text box is for keyboard entry of a sentence.
- **"Undo" Buttons:** These "Undo" buttons are conveniently located just below the map, so that the user can easily undo any undesired strokes. There are 2 buttons, one button for removing the last stroke the user drew on the map and another button for removing all strokes on the map.

3.2 Layout Design Rationale

The labels could have been placed next to the corresponding icons, as that would be clearer than listing the labels separately in a list box. However, placing the labels next to the icons introduced more clutter on the map display. If there are many landmarks near each other, it is very difficult to generate a legible layout of icons and labels. Listing the labels in a separate list box generated a neater and more pleasant appearance. Another approach would be to use different shapes or pictures to represent the icons. Due to the artistic complexity of generating these different types of icons, the simpler approach of representing them as squares was used instead. In general, the design assumes that all interactions with the interface, specifically the strokes and the mouse movements, should be intuitive and simple.

3.3 Type of input modes

Some information is more clearly and efficiently conveyed with a single input mode, while others are better conveyed using a combination. To make Voyager flexible and provide users with greater expressive power, multiple input modes were introduced in the interface. These input modes are speech and pen gestures. Although a keyboard entry box is provided, using it simultaneously with a mouse is cumbersome and inefficient. It is assumed that speech and text serve the same function, but speech

is more natural for humans during conversational interaction. The combination of pen gestures and speech gives users additional flexibility for expressing themselves. Multiple input modes allow the user to communicate in the most efficient and natural manner for a variety of queries.

3.3.1 Unimodal Verbal Input

Certain queries are most efficiently given in speech or text. Such queries are usually about known, unambiguous references. For example, if the user wants driving directions from Harvard to MIT, both of which are unique locations in the Boston area, the user can simply say,"Show me driving directions from Harvard to MIT." If the user wants to convey the same meaning using only gestures, the user could find the location of Harvard and MIT on the map, click on both of them, or draw an arrow originating from Harvard to MIT. Similarly, to do this multimodally, the user must click on the schools' locations and say, "How do I get from here to here?" Verbalizing the query is the most efficient method in this case.

Other actions which are most efficient verbally or by typing are:

- "Show me museums in Boston."
- "Give me an image of Boston."
- "Zoom in to the fourth one." (i.e., zoom in to 4th icon in the list.)

3.3.2 Unimodal Stroking Input

Queries that are most efficiently executed by solely stroking over the map or list box usually involve finding the label for an icon with a known spatial location or vice versa. For example, if the user is interested in finding out more about a particular icon on the map, but he doesn't know the name of the icon, he can click on the icon. The computer immediately knows the user's intentions and the icon and its corresponding label will be highlighted in the list box. However, if the user chooses to find out the name of the icon by verbally asking the system, he has to say something like, "What

is the name of the landmark two blocks south of the cross junction of Mass Ave and Prospect Street?" which is verbose and cumbersome. Even if the user executes this query multimodally, most likely by clicking on the icon and saying, "What is this icon?", the speech input is redundant. Other actions which the user can do most efficiently by stroking are:

- Move the mouse over the icon of interest (without any clicking) - icon and it's information in the list will be highlighted in magenta.
- Circle over a group of icons of interest - these icons and their information in the list will be highlighted in blue.

3.3.3 Multimodal Input

If an unlabeled road is on the map display and the user wants to know how to get to it, he can only refer to it by outlining or circling it and saying, "How do I get there from MIT?" or "How do I get there from here?" and stroke over his starting destination. Using only the speech or pen gesture modality to express the same query will be cumbersome to the user. If the user only spoke, he would need to say, "How do I get from the road right below Winchester Street and above State Street to MIT?" He must be as descriptive as possible to ensure that the road location is unambiguous. If he simply used gestures, he would need to draw a line or arrow starting from MIT going to the unknown road, though the meaning of a line may still be ambiguous. The line might imply focus on the landmarks of interest near it, which might lead to an incorrect interpretation. Gestures are most efficient when users want to refer to unknown spatial locations on the map. Sometimes, even when spatial locations are known, gestures may still be more desirable because the labels of the known locations may be difficult to pronounce or just lengthy to verbalize. For example, uttering, "How do I get from the Isabella Stewart Gardner Museum to the Charles Massachusetts General Hospital T-stop?", is more cumbersome than clicking on two locations on the map while saying, "How do I get from here to here?"

Other actions that the user can perform optimally multimodally are:

- User says, "How do I get there from MIT?" while clicking onto an icon or just moving the mouse over an icon.
- User says "Zoom in here," while clicking or moving the mouse to the desired location on the map.
- User says "How do I get from here to here?" and clicks two different locations on the map.
- User drew a couple of strokes. He wants to undo his last stroke and says "Undo."

3.4 Drawbacks

The design may bring about some drawbacks for the user. For example, while trying to make the map display look simple by separating the icon labels from the icons itself, the cognitive load of simultaneously looking at the list of labels and the map display to find the matching icon may be too high. Using a different font for the message windows and the list box may be uncomfortable to the eye. The presence of a keyboard entry box for debugging purposes may confuse users into using it, and stroking on the map at the same time as they type is not a natural mode of communication. Representing the icons as squares on the map is too simplistic. We could represent icons of different types with specialized symbols. For example for restaurants, we could represent it with a knife and a fork, for hotels we could use a house. These issues could be explored further in the future.

Chapter 4

Stroke Recognition

4.1 Overview

The GUI requires a gesture recognizer that understands strokes drawn on the map display. There are many gestures that can be drawn onto a map. Specific types have been chosen so that the gesture recognizer can recognize them easily and quickly. We took into consideration the types of gestures used during articulation and how these gestures conveyed an affirmative action.

4.2 Gesture Functions

Milota and Blattner [9] have categorized pen gestures that occur simultaneously with speech into 3 types:

- **Deictic Gestures** - pointing gestures that are used in conjunction with a spoken referent. Examples of such gestures:
  - Specific deictic - pointing to a specific object with the purpose of referring to that object, its function or attributes. In Voyager's case, this object could be a graphical icon or a textual display.
  - Generic deictic - pointing to a class of objects, their functions or attributes by pointing to an object in that class. Suppose there is a visual display

in a kiosk. A user may be pointing to a graphical icon in the display, but referring to all icons of the same type.

  - Mimetic Deictic - pointing with additional motion that selects among objects on a screen. For example, the motion of following a line with a pen to distinguish the line from other objects.

- **Attributive Gestures** - gestures that refer to attributes of objects. They can be used with or without deictic references. A user can put his pen over an object and drag the pen across the screen to indicate he wants to move it. Scratching, the motion of rapidly moving a pen back and forth over a displayed object, can indicate intensity together with the selection. A user may say, "make this planet blue," while scratching repeatedly on the planet icon. The repeated scratching action can intensify the color of the planet("Make it bluer").

- **Spaciographic-Pictomimic Gestures** - spaciographic gestures characterize an object by its shape. They can be used with inexact gestures to indicate approximate shape. Whenever possible verbal input would disambiguate the gesture. For example, a user may draw a triangle for the purpose of finding one or all triangles in a database or draw the correct shape of a curve by drawing another curve over it.

In the GUI, landmarks are represented as square-shaped icons on the map display. These icons cannot be reshaped or moved, so gestures referring to them are mainly deictic or attributive in nature. For referring to a single landmark, the most intuitive and simple gesture is the point click. Alternatively, a circle can be drawn around the icon. To refer to a group of landmarks, enclosing all of them in a circle is the simplest gesture. The map display also contains lines and curves to illustrate roads, highways and streets. To refer to a particular street, the user can simply draw a line along it. At times, landmarks of interest are placed along a street. A user can easily refer to all of them by drawing a line on it, as opposed to drawing an irregular circle around the street. With these considerations, a point click, a circle and a line are determined to be the most important gestures the recognizer should identify.

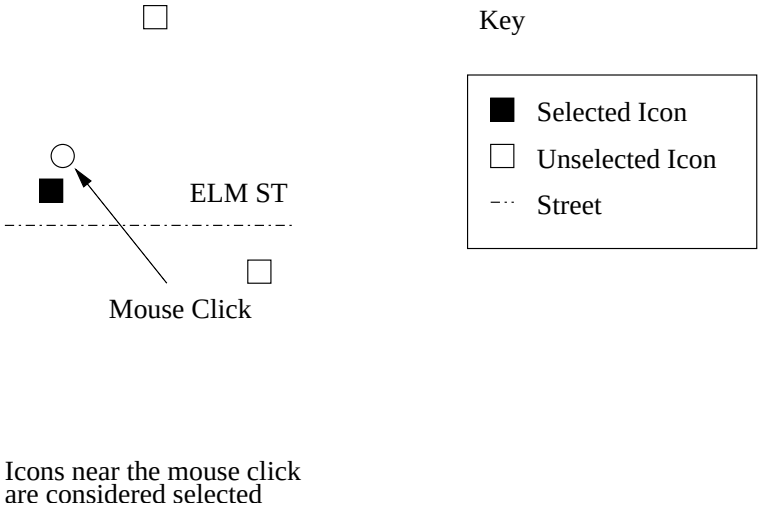

Figure 4-1: An example of a point click

4.3 Gesture Forms

4.3.1 A Point Click

There are two possible interpretations for a point click on the map. Users may be clicking on an icon of interest and requesting more details about that icon. This click becomes a selection of an icon, thereby focusing the attention of the dialog manager on the landmark represented by the icon. Figure 4-1 illustrates this idea. Users may also click a point on the map where there are no icons and say "Zoom in here." In this case, the point click refers to a specific geographic location not associated with a specific landmark. The point click serves as a deictic selection of a single referent.

4.3.2 A Circle

Sometimes users are interested in a group of landmarks displayed on the map. They will draw a circle around the icons and perhaps say, "Tell me more about these restaurants." The circle represents a deictic selection of a group of icons (see Figure 4-2). However, when users draw a circle with no icons in it, they may imply a region of interest and say, "Are there any restaurants over here?", which becomes an attributive

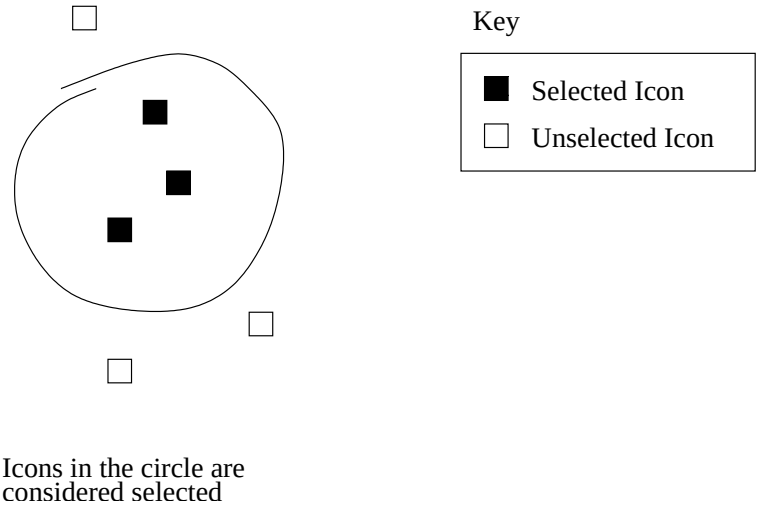

Figure 4-2: An example of a circle stroke and its interpretations

gesture. The dialog manager must be aware of this difference in stroke meaning to direct the conversation correctly. Certain ambiguities may arise. For example, an icon is circled but the user says, "Zoom in here." Depending on the size of the circle around the icon, the user may imply zooming to the icon of interest of just simply zooming into the encircled region.

4.3.3 A Line

In many instances during map navigation, we may be interested in landmarks or features along a section of a street or highway. Users may draw along a road of interest and ask, "Show me the restaurants along this road," or query, "What are the traffic conditions over there?". Figure 4-3 gives an illustration of this example. This line is interpreted as an attributive gesture for a road or a region of interest around the line. Others may draw a line to describe a trajectory on the map and ask, "How do I get from here to there?" In this case, the line indicates the source and destination location.

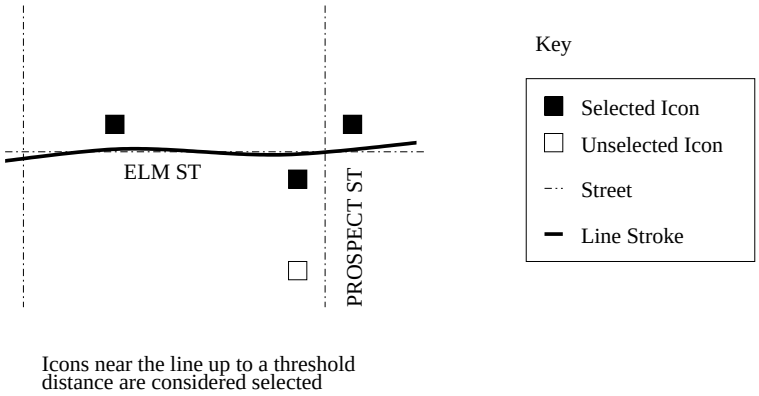

Figure 4-3: An example of a line stroke and its interpretation

4.3.4 Other Possible Gestures

Although point clicks, circles and lines are sufficient for deictic gestures, some other higher level strokes can convey more useful meanings. For example, a square may indicate the shape and boundary of a building, and an arrow may indicate a source and destination. We may recognize these gestures in the future.

4.4 Feature Based Recognition

A stroke is assumed to take place when the mouse is pressed and later released. Each time the mouse is released, a stroke is recognized. Since only 3 basic gestures are understood, a heuristic decision tree method of feature based recognition was developed. Most of these features are similar to features proposed by Rubine[12]. The features used in this work are:

- Area - the total area bounded by the minimum and maximum X and Y co-ordinates of the stroke.
- Ratio X - the ratio of the distance between the minimum X and maximum X-co-ordinate to the distance between the first point's X and the last point's X - co-ordinate.

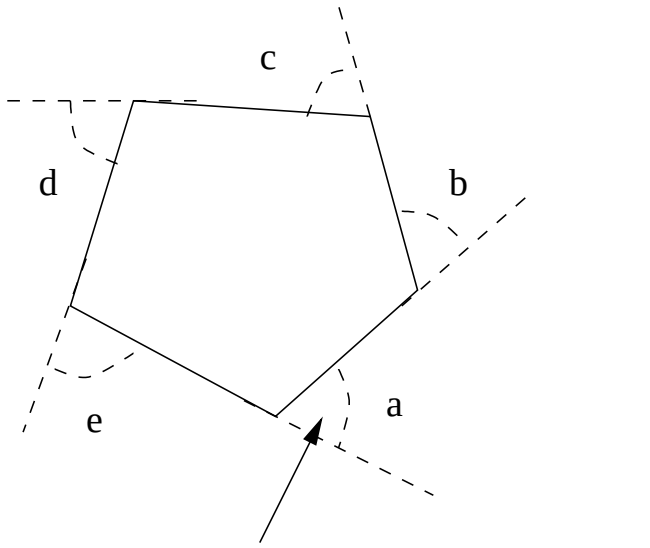

Figure 4-4: Sum of Side Angles

- Ratio Y - the same as Ratio X, except using Y-co-ordinates instead.
- Side Angle Sum - assume there is an N-sided polygon. At each vertex, we can compute the external angle(see Figure 4-4). The sum of these angles is the Side Angle Sum.
- Relative Distance Between First and Last Point - the distance between the first and last point normalized against the total length of the stroke.
- Convex Hull Violation times - the number of times the stroke violates the convex hull. A convex hull violation occurs when the original trajectory of the arc bends to the opposite direction, as shown in Figure 4-5. Each bend is considered as a count in the violation.

A diagram of the decision tree is shown in Figure 4-6. Each non-terminal node represents a question about one of the features. The terminal nodes represent the possible final decisions. For example, if the stroke satisfies condition A (i.e., is the rectangular area bounded by the stroke greater than 20^2 pixels?), the stroke moves

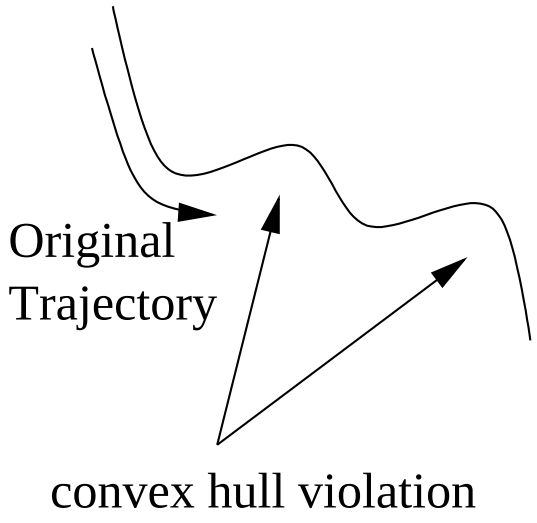

Figure 4-5: Convex Hull Violation

into the node labeled "POINT", and becomes recognized as a point. When "Ratio X" and "Ratio Y" are both more than 0.9 and the side angle sum is less than 0.9π, the stroke is determined to be a line.

Unlike the previous strokes, there are two possible ways a stroke can become a circle:

- When the side angle sum is between 1.8π and 2.3π and the relative distance between the 1st and last point is less than 0.5.

- When the side angle sum is between π and 2.5π and the convex hull violation count is less than 3.

The circle has multiple possibilities because users do not always draw a perfect, uniformly curved circle and we have to cater to different variations of how users may draw a circle. As shown in Figure 4-7, visually speaking, stroke 1 and stroke 2 should be accepted as a circle stroke, but stroke 3, although it does make an enclosure, has 3 instances of convex hull violations, and seems like an unintuitive shape for a circle.

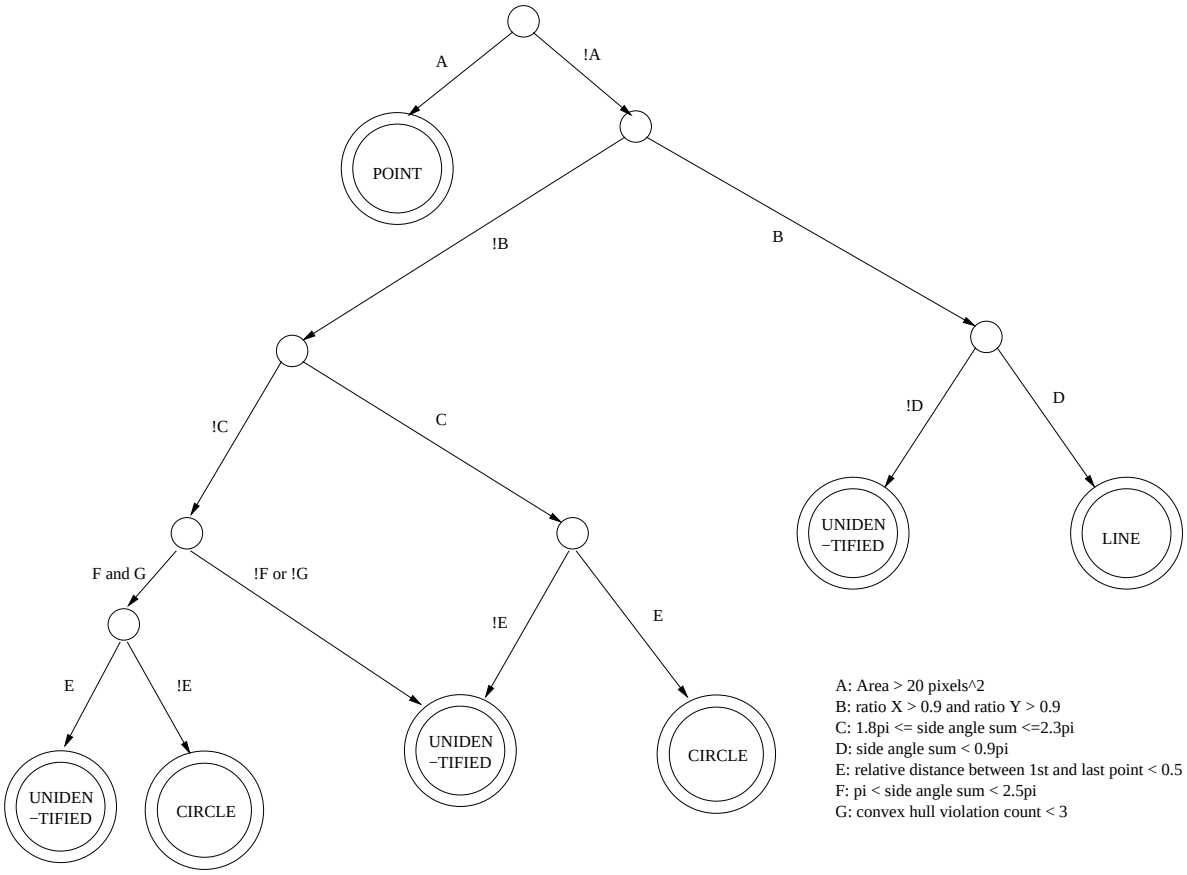

Figure 4-6: The Decision Tree of the Gesture Recognizer.

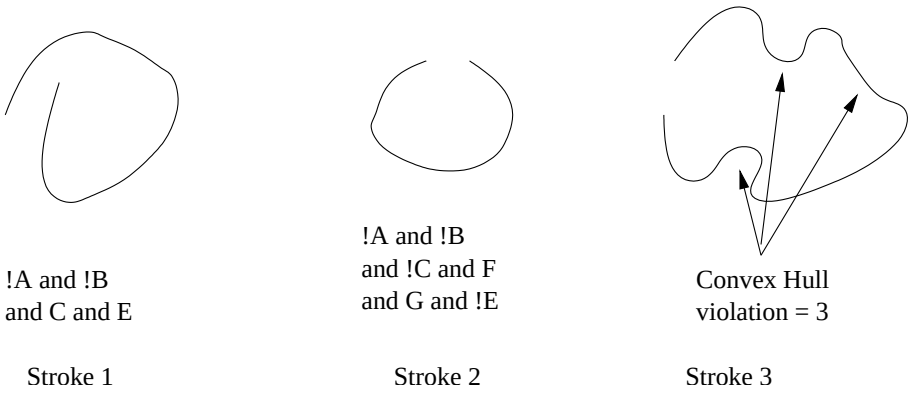

Figure 4-7: Multiple types of a circle and a non-circle stroke.

4.5 Drawbacks

The main drawback of the gesture recognizer is that it is too simplistic. Many potentially important and useful gestures that humans could use for human-computer interaction cannot be recognized and hence cannot be used by the interface. As a start, these three gestures are sufficient, providing the deictic and attributive information with simultaneous speech. Sometimes strokes that look conspicuously like a circle or a line are recognized incorrectly. In the future, our heuristic method of stroke recognition will be replaced with a discriminately trained decision tree or a statistical approach such as Gaussian mixture models. The recognizer will then be trained off data from people drawing circles, lines and points.

Chapter 5

Multimodal Database

A means of storing the mouse events in the GUI is required, so that the multimodal context resolution could use its information for deictic resolution. However, recording all the low level events, such as the x-y co-ordinates of the mouse position at all time instances, is not an efficient method. Such events do not explain what the mouse or the stroke is doing. There should be gesture interpretations of these events. At the same time, these gesture interpretations should be simple and straightforward in meaning such that the multimodal context resolution component can easily combine it with the semantic information extracted from the speech. In view of these considerations, all mouse events are associated with 4 higher level interpretations, namely Icon Select, Icon Highlight, Geographic Selection and Geographic Location.

5.1 Icon Selection

The most affirmative mouse events are clicks and strokes (which involve pressing the mouse and dragging it over the map display). If any icons are selected by the gesture, those icons are considered to be selected. This change in focus is visually fed back to user by highlighting the selected icon in the map display. The possible interpretations are shown in Figure 5-1. These interpretations include icons bounded by a circle, icons near a line and icons near a mouse click, are referred to as "Icon Selection" events.

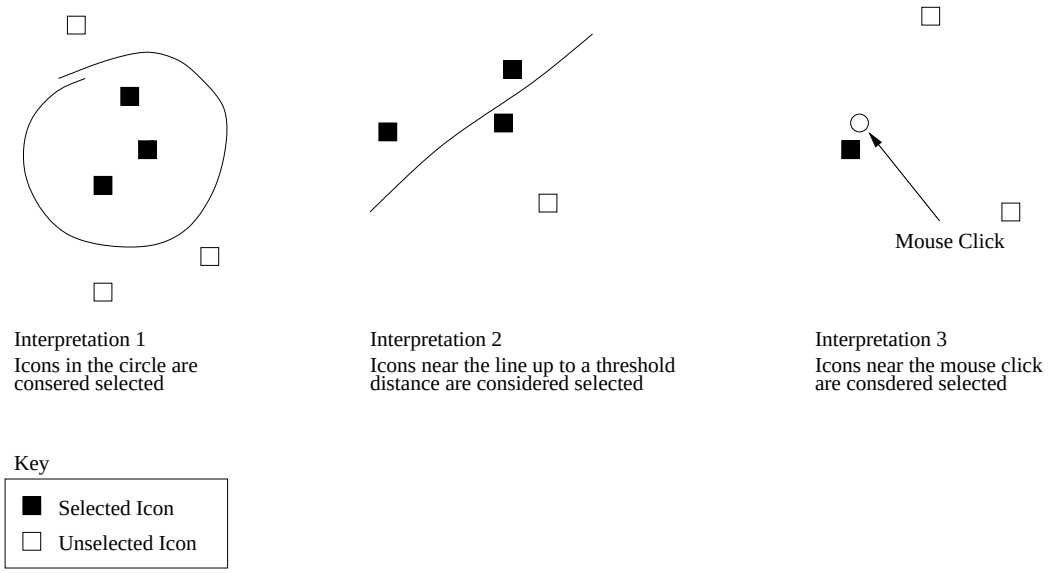

Figure 5-1: Example gestures and their interpretations as "Icon Selection" events.

5.2 Icon Highlight

When a user moves his mouse near an icon but does not select it, the icon changes color to inform the user that the GUI has placed its current focus on the icon. However, since no mouse clicks are involved, the action is not considered to be affirmative and cannot be considered a definite icon selection. Hence, it is classified as an "Icon Highlight" event.

5.3 Geographic Selection

Sometimes, the user may express interest in a specific region or location on the map. He may click a spot on the map or draw on a circle on the region of interest where no icons are present. In such events, the bounding box of the stroke provides the geographic region of the user's interest. Since such events involve dragging or clicking the mouse, it is assumed to be an affirmative event and is classified as a "Geographic Selection" event.

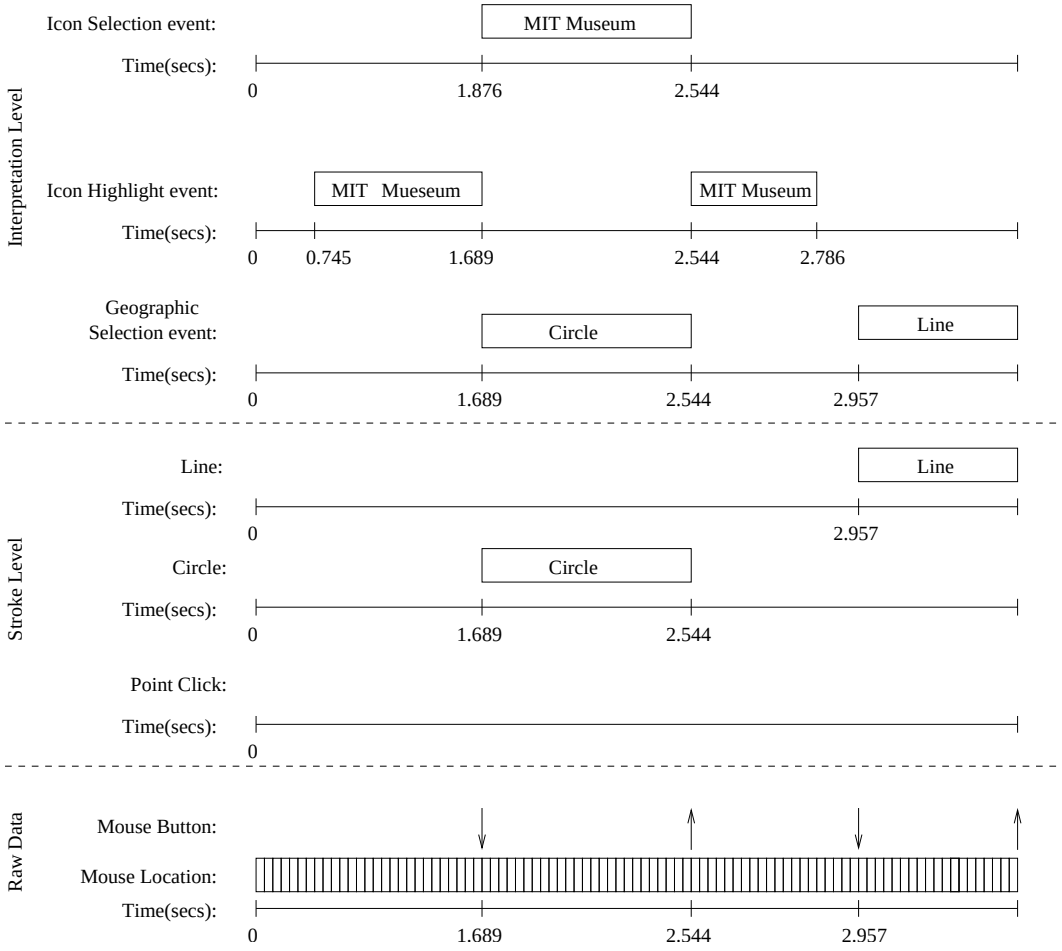

Figure 5-2: Timing diagram of the interpretations by the database and their correlation to the strokes and mouse location

5.4 Geographic Location

A user may just move the mouse to a location of interest, without any dragging or clicking, with no nearby icons. This action is not as affirmative as a "Geographic Selection" event, but the geographic location represented by the mouse location still provides useful information. Hence this event is interpreted as "Geographic Location."

5.5 Recording the events

As shown in Figure 5-2, there is a hierarchy associated with converting the mouse events into interpretations. At the lowest level is the mouse location and mouse button action. At the stroke level, the gesture recognizer classifies the mouse events into a point, line or circle. Depending on the proximity of the icons, the multimodal database interprets and categorizes these strokes into "Icon Selection", "Icon Highlight", "Geographic Selection" and "Geographic Location" Events. Note that the "Geographic Location" event is simply the Mouse Location and hence it is not displayed as a separate interpretation from mouse location. In this figure, when the mouse was moving over the map from 0.745 to 1.689 seconds (without any clicking), MIT Museum was highlighted on the map. This event is interpreted as an "Icon Highlight" event. However, when the mouse button was pressed, dragged and then released from 1.689 seconds to 2.554 seconds, a circle was drawn and the MIT Museum icon was selected. This event is interpreted as an "Icon Selection" event. Because the user may also alternatively be referring to the geographic region near the MIT Museum (and not the museum itself), the circle is also registered as a possible "Geographic Selection" event. From 2.957 seconds and beyond, a line was drawn, but since no icon was in close proximity of the line, the event becomes interpreted as a "Geographic Selection" event.

Once the mouse events and strokes are correctly interpreted, they are stored in the multimodal database. Each event stores different information. In the case of "Icon Selection" events, information like the start times and end times of the stroke or click,

the name and description of the landmark, and the type of stroke selection are stored into the database. For "Icon Highlight" events, the start times of the highlighting action, the name and description of the landmark are stored instead. For "Geographic Selection" events, the stroke type, and the bounding box of the stroke is recorded.

Chapter 6

The Multimodal Context Resolution Mechanism

All the speech infrastructure for Voyager is provided by the Galaxy system[14]. As shown in Figure 2-2, when the user speaks to the system, the utterance sends the speech waveform to the speech recognizer, which creates a word graph of hypotheses. This information is then dispatched to the natural language unit, where the word graph is parsed by the TINA[13] natural language understanding system. The best hypothesis, encoded in a semantic frame representation, is sent to the context resolution server. This server resolves unknown references in this semantic frame by incorporating all implied information from the user's semantic frames in previous turns of the dialog, which are stored in a history[3]. References that needed to be resolved can be words like "here", "there", or "this" which are contextual pronouns. The resolved semantic frames are passed to the dialog manger, which determines the appropriate response to the user. In the new version of Voyager, this method of resolution may generate some errors. Suppose the user says,

> Utterance 1: "Hi, show me the location of the MIT Museum." (no mouse movement on the GUI)
>
> Utterance 2: "I am going there tomorrow, can you show me the driving directions from my home?" (2nd utterance) (no mouse movement on the GUI)

From the first utterance, context resolution is aware that the topic in focus is the MIT museum. In the 2nd utterance, it can use this topic to resolve the unknown reference "there." However, if the user says,

> Utterance 1: "Hi, show me the location of the MIT Museum." (no mouse movement)
>
> Utterance 2: "I am going there tomorrow, can you show me the driving directions from my home?" (a mouse click is registered on the map, although it maybe near the icon representing MIT Museum, the icon is not selected)

In this case, the user meant "there" as the location he had clicked on the map. This pronoun now serves a deictic instead of an anaphora. The context resolution cannot correctly resolve discourse this deictic with a topic from the 1st utterance. Since complex mouse events on the map and the GUI need to be taken into consideration, a pre-defined preference mechanism was developed to resolve these different types of references.

6.1 Multimodal Resolution Mechanism

Icons associated with mouse events occurring while a deictic word is uttered are treated as the deictic reference. However, many mouse events may occur at the same time, and a priority list is needed to choose the appropriate one. As mentioned in Chapter 5, the mouse events have been categorized as "Icon Select", "Icon Highlight", "Geographic Selection" and "Geographic Location". "Icon Selection" events have the highest priority while "Geographic Location" events have the lowest priority. Figure 6-1 explains the decision process in resolving the deictic. The semantic frame passed to the discourse server contains the start time and end time of this deictic phrase, and this timing info is used to determine if any viable mouse events are present to resolve the deictic expression. A diagram illustrating the different streams of mouse events is shown in Figure 6-2. The user says, "How do I get there?" and assuming no speech recognition errors, the semantic frame parsed by the natural language unit has

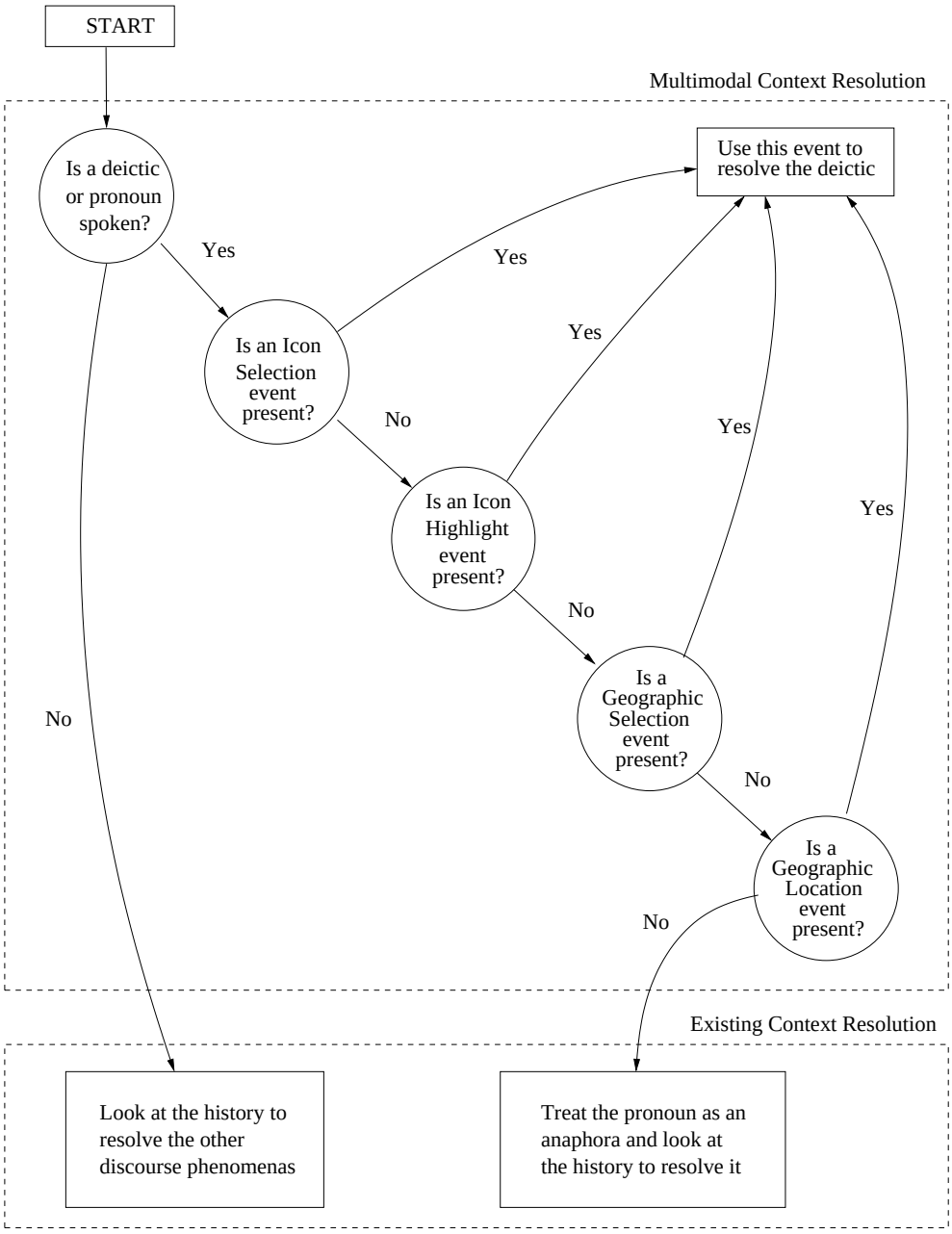

Figure 6-1: Decision tree of multimodal resolution

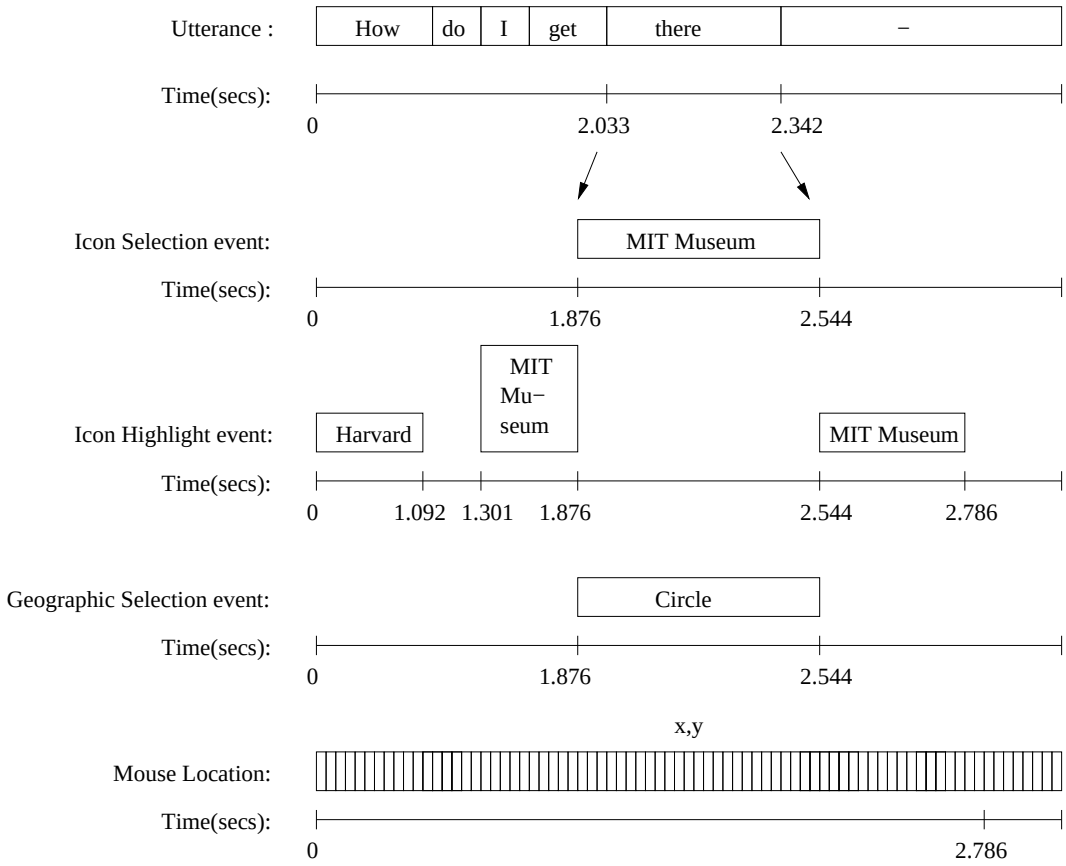

Figure 6-2: An example of multimodal deictic resolution

the exact same phrase. An "Icon Selection" event was present within the start time (2.033 secs) and end time (2.342 secs) of the deictic "there," and the MIT Museum was selected. Since "Icon Selection" events have the highest priority, this event is used to resolve the deictic. Oviatt[10] wrote that not everyone makes a selecting gesture strictly within the time range at which the deictic is uttered, so any "Icon Selection" event that occurred within 1.7 seconds before or after the deictic are accepted. This time range may be a little too stringent and will be further explored in the future.

If two or more unique "Icon Selection" events were registered instead at the same time, the context resolution would be unable to resolve the deictic to a definite reference. It will pass this indeterminate result to the dialog manager, which could possibly ask the user, "Which one are you referring to?" This will be implemented in the future.

Chapter 7

User Experiments

7.1 Motivation

The integration of speech and pen gestures has generated much interest in the research community. The Quickset system[2] integrates speech with pen input that includes drawn graphics, symbols, gestures, and pointing. The Quickset system uses a semantic unification process to combine the meaningful multimodal information carried by two input signals. Quickset uses one modality to disambiguate meaning in the other. Johnston recently developed MATCH (Multimodal Access To City Help)[7], a mobile multimodal speech-pen interface to restaurant and subway information for New York City, which uses finite state transducers for the multimodal integration process. In these systems, using multimodal integration to resolve pronouns like "this", "there" and "here" is a common process. Depending on the context of the systems, such words can function either as deictics or as anaphoras. The different techniques in multimodal resolution serve to determine the function of such words according to their context.

One possible cue to determine the function of pronouns or other referential words is detecting the presence of gestures. The presence of a gesture during an utterance implies that a deictic word was more likely spoken. With this association, if the speech recognizer has poor performance recognizing deictic words, boosting the recognition of such words with the presence of gestures seems plausible. For this thesis, different

	Total
{5.365} zoom {4.672} in {5.035} right {−7.827} {−5.935}	1.31
{5.377} zoom {4.642} in {5.007} Brighton {−8.068} {−5.967}	0.991
{5.345} zoom {4.653} in {5.016} right {−11.449} here {−5.981}	−2.417
{5.390} zoom {4.662} in {5.025} ride {−12.367} here {−4.064}	−3.225

Figure 7-1: N-best hypotheses generated by the SUMMIT speech recognizer with the actual spoken utterance underlined.

boosting mechanisms have been applied and analyzed to see how well each could boost the performance of the speech recognizer.

7.2 Scoring with Speech Recognition

Currently, all user's utterances are processed and recognized by the SUMMIT speech recognizer[4], which generates an N-best list of hypotheses of possible sentences. Suppose a user says, "Zoom in right here," while clicking on a landmark on the map display of the Voyager's graphical interface. The N-best list of hypotheses generated from the recognizer is given in Figure 7-1. The number to the right of each word is the confidence score of the word [6]. This word score is expressed as :

$$Word\ Score \;=\; \log \frac{P(word\ is\ correct)}{1 - P(word\ is\ correct)} \tag{7.1}$$

The total score of each hypothesis is the sum of all these numbers. As shown in this figure, the correct hypothesis only has the third best score. However, if a point click or a stroke gesture was present during the deictic utterance, the score of the deictic word "here" could be boosted with an additional score from the gesture, and hence raise the total score of the hypothesis containing this word. To ensure clarity in the rest of this chapter, here is some terminology:

Deictic Utterance - utterance that contains deictic references.

Deictic Gestures - the pointing/selecting gesture.

Deictic Hypothesis - the hypothesis that contains one or two deictic words. This hypothesis is only one of the entries on the N-best list.

7.3 Boosting Recognition Scores with Gestures

Any motion of the mouse over the map display is considered a gesture. The user is assumed to be clicking, dragging or just moving the mouse over the map display only when he is interested in some landmark, street, or area. As mentioned in Chapter 5, these gestures are divided into three types of events: "Icon Selection", "Icon Highlight", "Geographic Selection" and "Geographic Location." For "Icon Selection" and "Geographic Selection" events, each individual event is considered a single gesture. For "Icon Highlight" events, when the mouse is close to an icon, then the icon is highlighted and it is registered as an event. However, since the GUI registers events whenever the mouse is moved, many "Icon Highlight" events are recorded. If the user intends to highlight a specific icon, it is assumed that the mouse will be moving continuously near the icon for at least three time frames, which results in at least three consecutive instances of the same icon highlight event. Hence, three consecutive highlight events for the same icon constitutes one gesture. The same applies for "Geographic Location" events, except that five consecutive "Geographic Location" events constitute one gesture. If the mouse is on the map display, no icons are nearby and it is not moving at a given time, no events are interpreted.

Different methods have been proposed for boosting the recognition scores with these gestures. In general, for the different boosting methods, the total score is expressed as:

$$total\ score = recognition\ score + boost\ weight \times boost\ score$$

where the variations occur in the boost score.

- Method A - boosting either with number of deictic words in the deictic hypothesis or the number of gestures.

  For each entry in the hypotheses, (in pseudocode)

  if number of gestures > number of deictic words in hypothesis

boost score = number of deictic words in hypothesis

else

boost score = number of deictic gestures

compute total score

- Method B - boosting based on the number of deictics in hypothesis, the number of gestures and the type of gesture.

  For each entry in the hypotheses, (in pseudocode)

  if a gesture is present and there is a deictic hypothesis,

  boost score = max(gesture type factor) × number of gestures × number of deictic words in hypothesis

  else

  boost score = 0

  compute total score

  gesture type factor : 1 (if "Icon Selection")
  0.5 (if "Icon Highlight")
  0.25(if "Geographic Location" or "Geographic Selection")

- Method C - boosting based on the number of hypothesized deictics and the number of gestures only.

  For each entry in the hypotheses, (in pseudocode)

  if a gesture is present and there is a deictic hypothesis,

  boost score = number of gestures × number of deictic words in hypothesis

  else

  boost score = 0

  compute total score

- Method D - boosting based on the number of hypothesized deictics only, regardless if a gesture was present.

For each entry in the hypotheses, (in pseudocode)

boost score = number of deictic words in hypothesis

compute total score

7.4 Experiment Procedure

Data on user interaction with Voyager was collected. Users were instructed to accomplish the following tasks, either by speech or pen or both simultaneously:

- Display a map of a city in the Boston area.
- Display a map containing a particular type of landmark(e.g. museums, historic sites, public gardens etc).
- Zoom in and out of one of the landmarks or selected items.
- Retrieve driving directions from one displayed landmark to another.
- Ask for traffic information for a major road or highway visible on the map.

Users were to behave as if the system was performing perfectly even if it generated erroneous responses. While the users were performing these tasks, the N-best list generated by the recognizer for every user's utterance and the gestures produced on the map were collected.

7.5 Results

There were a total of 506 utterances collected from 21 users. The average number of utterances per user was 24.1. 149(29.45%) of these utterances contained deictic references. The average number of words spoken per sentence was 5.6.

The statistics are tabulated in Figure 7.1. As shown in the table, 95 deictic references were not correctly hypothesized as the 1st entry in the N-best list. This is an all or none result over all deictic references. Of these incorrect hypotheses, 25 of them (case C1) had deictic references hypothesized in other entries of their N-best

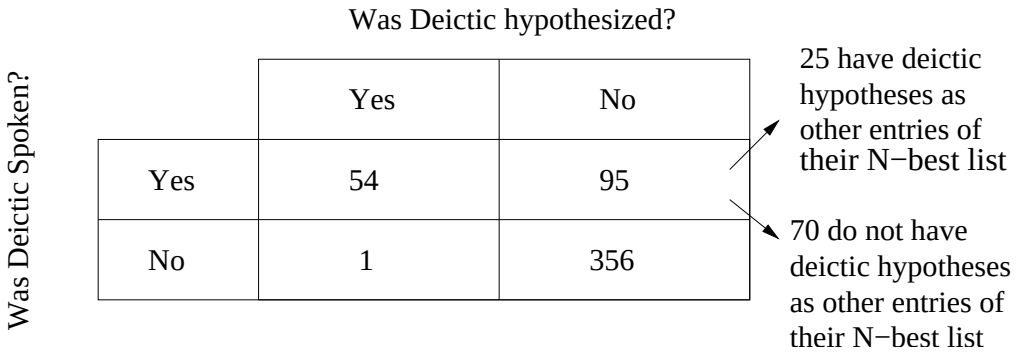

Table 7.1: Results tabulated according to deictic utterances and deictic hypotheses

list, while 70 of them (case C2) had none. The scores of case A, B and C1 could be boosted as they contain at least one deictic hypothesis in one of their N-best list entries. Using the methods mentioned above, the overall score of the hypotheses and the Word Error Rates of all the utterances and deictic utterances only were computed and the results are shown in Table 7.2.

When all utterances were taken into consideration, the word error rate was 38.9% (1027 errors) without boosting. It improved to 37.1% (979 errors) after boosting using the best method A. The relative word error rate reduction is only 4.7%, which is not very significant. However, when only deictic utterances were taken into consideration, the word error rate was 35.3%(177 errors) without boosting. It improved to 25.7%(129 errors) after boosting using the best method A. The relative word error rate reduction was 27.2%, a much more significant improvement.

The results of applying different boost weights on different methods is shown in Figure 7-2. The word error rate decreases monotonically to an asymptote as the boost weight increases. This shows that a high value of boost weight will not hurt the word error rate, and that the best boost weight for each method can be applied. Method A and D had word error rates of 37.1% while method B and C had word error rates of 37.2%. This difference is of little significance and implies that methods A, B, C and D are very similar.

	Word Error Rates (%)	
	Deictic Utterances	All Utterances
No Boosting	35.3	38.9
Method A	25.7	37.1
Method B	26.3	37.2
Method C	26.3	37.2
Method D	26.3	37.1

Table 7.2: Gesture boosting and its impact on word error rate for all utterances and deictic utterances

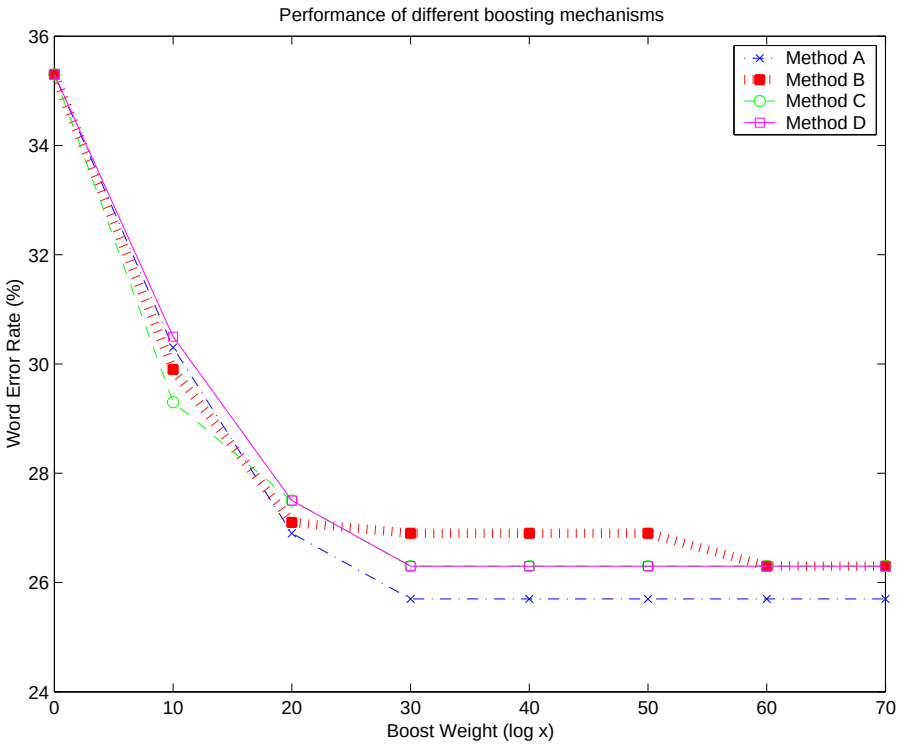

Figure 7-2: Word error rates at varying boost weights for deictic utterances

7.6 Conclusion

In general, these results show that gestures can boost the recognition score of deictic words and hence improve the word error rate. When applied to all utterances, the improvement in word error rate seems insignificant(a relative word error rate reduction of 4.7% for the best method). However, the utterances that could be boosted were only the ones where gestures were present simultaneously with a deictic hypothesis, which takes up approximately 30% of all utterances. For these types of cases, the best method reduced the word error rate by 27.2%, which is a significant amount.

All the methods of boosting the recognition scores were dependent on the presence of a deictic hypothesis in one of the entries of the N-best hypotheses. As shown in Table 7.1, there were 70 deictic utterances(Case C2) that had no deictic hypotheses in the N-best list, which was almost three times more than the deictic utterances that contain them(Case C2). All the methods had a major flaw: if no deictic words were in the utterance, but a deictic was an entry of the N-best list, and a gesture was present during the utterance, the incorrect deictic hypothesis will be boosted to the top score and increase the word error rate. In this data set, there was one such instance in Case D. It could be argued that since such instances have a much lower occurrence(1 utterance) than instances like case C1(25 utterances), the overall word error rate will still improve. However, during the data collection, almost all users were observed to be in an attentive state when interacting with the system. One might speculate that these users felt a huge cognitive load while facing the computer screen, but the user who contributed the special case in case D was actually interacting with the computer jovially, which should be the ideal conditions for human-computer communication. If users were interacting with the system in a more natural and relaxed manner, more incidences of this case may arise. This experiment also has to take into account that all 21 users were graduate students or researchers in the MIT Spoken Language Systems research group. These users are experienced in using speech-based systems, hence the data collected may not accurately reflect the patterns of behavior of the general population.

When computing the word error rates for cases where gestures had occurred and a deictic hypothesis exists as one entry in the N-best list, method A showed a 27.2% relative word error reduction rate, while method D showed a 25.5% relative word error reduction rate. This implies that method D had boosted the recognition scores of cases where there were no gestures but words like "there" was spoken. In such instances these words function as anaphoras instead. For example, the user could have said, "Show me the map of MIT." in the first utterance and later said, "How do I get there from Harvard university?", in the second utterance.

The different methods of boosting produced insignificant differences for all utterances. Methods A and D had an overall word error rate of 37.1% which is 979 errors while methods B and C had an overall word error rate of 37.2% which is 982 errors, three more than method A and D. Nevertheless, increasing the boost weights for all methods have resulted in a consistent monotonic decrease in the word error rate. All the word error rates reach an asymptote when the boost weight reaches a high value of 30-60.

Overall, these results show that although boosting the speech recognition scores with gestures improves the overall word error rate by an insignificant amount of 1.8%, the word error rate of multimodal utterances was actually reduced by a significant amount(27.2%). Although the gestures are only useful for the disambiguating deictic words like "here", "there", "this" and "that", which constitutes a small class of words in all the possible English words that could be spoken, these words may be very common in multimodal systems.

Chapter 8

Conclusions and Future Work

8.1 Summary

Pen-based pointing and sketching gestures have been added to the Voyager system. The graphical interface was modified to incorporate pen and mouse gestures on the map display and a stroke recognizer was developed to recognize these gestures. These gestures and their contextual meanings were recorded in a new multimodal database. To resolve unknown speech references using the contextual events in the database, a new multimodal context resolution mechanism was developed to augment the existing context resolution. Voyager is now capable of displaying maps, landmarks, historic sites, restaurants and public gardens and users can successfully query for driving directions from one landmark to the other either multimodally or just using speech. User experiments were conducted to see how gestures can boost the speech recognition score of deictic words spoken by users.

8.2 Future Work

Currently, users can only make successful queries to Voyager only if they speak to it or speak to it while drawing on the display. In addition, the pen gestures users can draw on Voyager are essentially deictic ones. Oviatt[10] wrote that since deictic gestures account for only 14% of all spontaneous multimodal utterances, systems that

specialize in processing speaking-pointing relations will have limited practical use. This highlights the importance of systems being able to function both unimodally and multimodally. As such, more functionality for gestures should be provided such that the user can make queries by gestures successfully. As mentioned previously in Chapter 4, spaciographic and attributive gestures should be added to the system. A user could draw a square defining the boundaries of a building on the map. A line initially drawn to indicate the source and destination, could be dragged or stretched when the mouse is over it and the mouse button held down. Relations between different shapes can be established such that higher level objects can be recognised and understood. For example, if the user draws a simple flower, it may imply interest in flower shops available on the displayed map.

With the addition of more complex gestures from pen strokes, a complex meaning representation for multimodal integration has to be developed. Apart from resolving deictic references, context resolution will need a new framework that can combine semantic and gesture meaning. Perhaps graphical models[11] can be developed to structure the dependence of different gesture meanings with different topics in speech.

8.3 Conclusion

The introduction of pen-based gestures to Voyager has offered users greater expressive power and naturalness. The Galaxy Architecture was developed to enable developers to add new server capabilities easily without disrupting the existing speech framework, but figuring out the integration and synchronization requirements for combining the speech mode and the gesture mode strategically into a dynamic whole system has been a great challenge. The development of this multimodal integration framework can serve as a model for other modalities to be added to Galaxy in the future. From user experiments, utterances with deictic gestures showed good improvements in recognition if boosted by gestures. This shows that the presence of different modalities can help disambiguate the other. Although there are not much functionality provided to pen gestures only, this new integrated multimodal Voyager system has supported

entirely new capabilities that have not been supported at all by the previous Voyager system and more work could be done to enhance this multimodal capability. Adding new modalities will make Voyager a more robust multimodal system.

Bibliography

[1] R.A. Bolt. Put that there, voice and gesture at the graphics interface. *ACM Computer Graphics 14*, 1980.

[2] P. Cohen, M. Johnston, and D. McGee. Quickset: Multimodal interaction for distributed applications. In *Fifth ACM International Multimedia Conference*, New York, NY, 1997. ACM Press, NY.

[3] E. Filisko. A context resolution server for the GALAXY conversational systems. Master's thesis, MIT Department of Electrical Engineering and Computer Science, Cambridge, MA, May 2002.

[4] J. Glass. A probabilistic framework for segment-based speech recognition. *Computer Speech and Language*, 17(2-3):137–152, 2003.

[5] D. Goddeau, E. Brill, J. Glass, C. Pao, M. Phillips, J. Polifroni, S. Seneff, and V. Zue. Galaxy : A human language interface to online travel information. In *International Conference of Speech and Language Processing*, Yokohama, Japan, 1994.

[6] T. Hazen, S. Seneff, and J. Polifroni. Recognition confidence scoring and its use in speech understanding systems. *Computer Speech and Language*, 16:49–67, 2002.

[7] M. Johnston, S. Bangalore, A. Stent, G. Vasireddy, and P. Ehn. Multimodal language processing for mobile information access. In *International Conference of Speech and Language Processing*, Denver, Colorado, 2002.

[8] D. B. Koons, C.J. Sparrell, and K.R. Thorisson. *Intelligent Multimedia Interfaces*, chapter Integrating simultaneous input from speech, gaze and hand gestures., pages 257–276. MIT Press, Menlo Park, CA, 1993.

[9] A. D. Milota and M. M. Blattner. Multimodal interfaces using pen and voice input. In *International Conference on Systems, Man and Cybernetics*, Canada, Vancouver, October 1995.

[10] S. Oviatt. Ten myths of multimodal interaction. *Communications of the ACM*, 1999.

[11] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*, chapter 3, pages 77–131. Morgan Kaufmann, Los Angeles, CA, 1998.

[12] D. Rubine. Specifying gestures by example. *Computer Graphics*, 25(4), 1991.

[13] S. Seneff. TINA: A natural language system for spoken language applications. *Computational Linguistics*, 18(1):61–86, 1992.

[14] S. Seneff, E. Hurley, R. Lau, C. Pao, P. Schmid, and V. Zue. Galaxy-2: A reference architecture for conversational system development. In *International Conference of Speech and Language Processing*, Sydney, Australia, 1998.

[15] V. Zue. JUPITER: A telephone-based conversational interface for weather information. *Speech and Audio Processing*, 8(1), 2000.