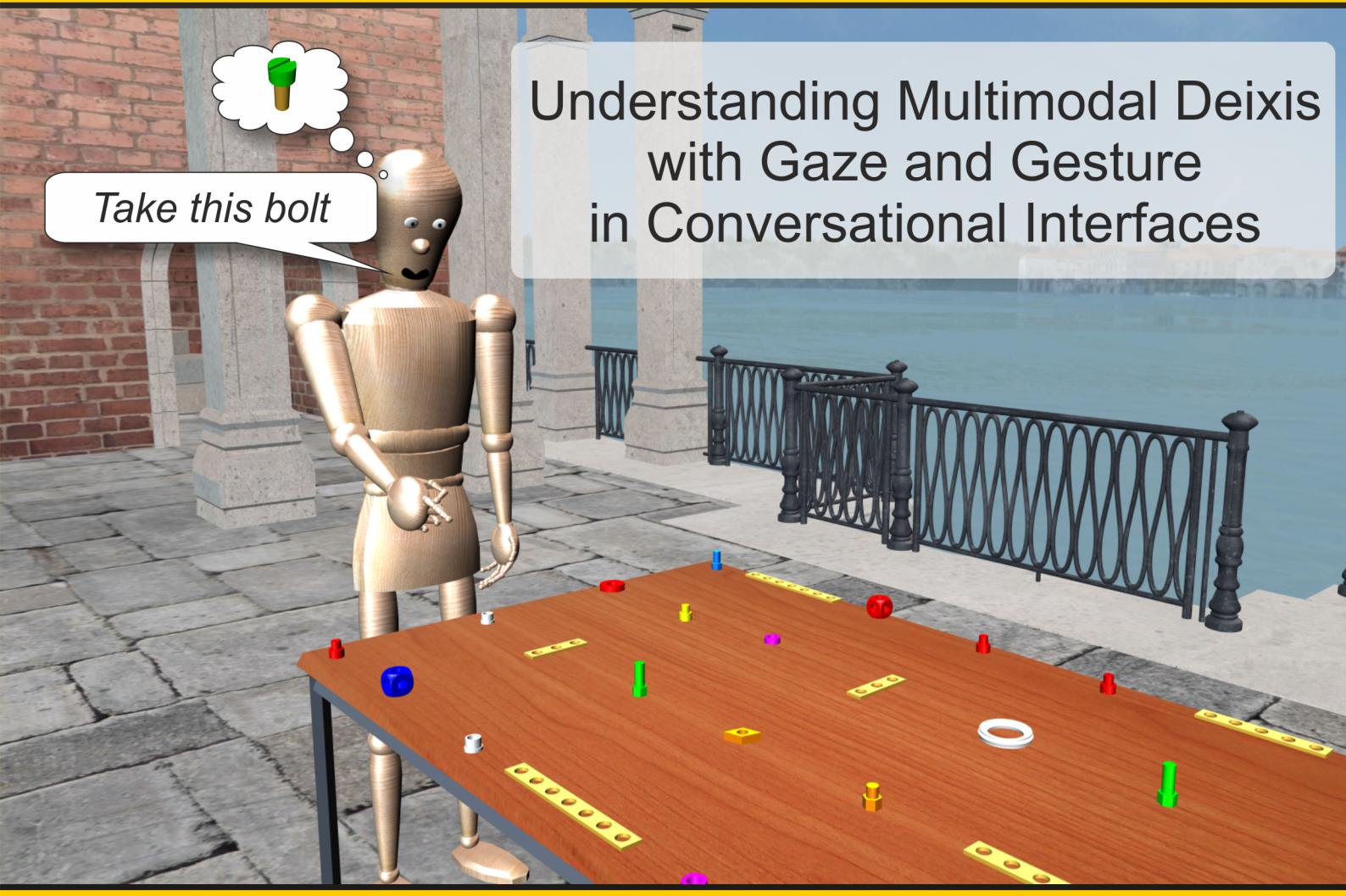


Understanding Multimodal Deixis with Gaze and Gesture in Conversational Interfaces

Thies Pfeiffer
April 28, 2010

Understanding Multimodal Deixis with Gaze and Gesture in Conversational Interfaces

Thies Pfeiffer
A.I. Group
Faculty of Technology
Bielefeld University
P.O. Box 10 01 31
D-33501 Bielefeld
Germany
email: tpfeiffe@techfak.uni-bielefeld.de

This dissertation has been approved by the Faculty of Technology at Bielefeld University to obtain the academic degree of a Doctor rerum naturalium (Informatics).

Dean of the faculty: Prof. Dr. Jens Stoye
First reviewer: Prof. Dr. Ipke Wachsmuth
Second reviewer: Prof. Dr. Hannes Rieser

Submission of the thesis: April 28, 2010
Day of the disputation: October 8, 2010

The official print version has been printed on age-resistant paper according to DIN-ISO 9706.

Summary

When humans communicate, we use deictic expressions to refer to objects in our surrounding and put them in the context of our actions. In face to face interaction, we can complement verbal expressions with gestures and, hence, we do not need to be too precise in our verbal protocols. Our interlocutors hear our speaking; see our gestures and they even read our eyes. They interpret our deictic expressions, try to identify the referents and – normally – they will understand. If only machines could do alike.

The driving vision behind the research in this thesis are multimodal conversational interfaces where humans are engaged in natural dialogues with computer systems. The embodied conversational agent Max developed in the A.I. group at Bielefeld University is an example of such an interface. Max is already able to produce multimodal deictic expressions using speech, gaze and gestures, but his capabilities to understand humans are not on par. If he was able to resolve multimodal deictic expressions, his understanding of humans would increase and interacting with him would become more natural.

Following this vision, we as scientists are confronted with several challenges. First, accurate models for human pointing have to be found. Second, precise data on multimodal interactions has to be collected, integrated and analyzed in order to create these models. This data is multimodal (transcripts, voice and video recordings, annotations) and not directly accessible for analysis (voice and video recordings). Third, technologies have to be developed to support the integration and the analysis of the multimodal data. Fourth, the created models have to be implemented, evaluated and optimized until they allow a natural interaction with the conversational interface.

To this ends, this work aims to deepen our knowledge of human non-verbal deixis, specifically of manual and gaze pointing, and to apply this knowledge in conversational interfaces. At the core of the theoretical and empirical investigations of this thesis are models for the interpretation of pointing gestures to objects. These models address the following questions: *When* are we pointing? *Where* are we pointing to? *Which* objects are we pointing at? With respect to these questions, this thesis makes the following three contributions:

First, *gaze-based interaction technology for 3D environments*: Gaze plays an important role in human communication, not only in deictic reference. Yet, technology for gaze interaction is still less developed than technology for manual interaction. In this thesis, we have developed components for real-time

tracking of eye movements and of the point of regard in 3D space and integrated them in a framework for D*eictic* R*eference* I*n* V*irtual* E*nvironments* (DRIVE). DRIVE provides viable information about human communicative behavior in real-time. This data can be used to investigate and to design processes on higher cognitive levels, such as turn-taking, check-backs, shared attention and resolving deictic reference.

Second, *data-driven modeling*: We answer the theoretical questions about timing, direction, accuracy and dereferential power of pointing by data-driven modeling. As empirical basis for the simulations, we created a substantial corpus with high-precision data from an extensive study on multimodal pointing. Two further studies complemented this effort with substantial data on gaze pointing in 3D. Based on this data, we have developed several models of pointing and successfully created a model for the interpretation of manual pointing that achieves a human-like performance level.

Third, *new methodologies for research on multimodal deixis in the fields of linguistics and computer science*: The experimental-simulative approach to modeling – which we follow in this thesis – requires large collections of heterogeneous data to be recorded, integrated, analyzed and resimulated. To support the researcher in these tasks, we developed the I*nteractive* A*ugmented* D*ata* E*xplorer* (IADE). IADE is an innovative tool for research on multimodal interaction based on virtual reality technology. It allows researchers to literally immerse into multimodal data and interactively explore them in real-time and in virtual space. With IADE we have also extended established approaches for scientific visualization of linguistic data to 3D, which previously existed only for 2D methods of analysis (e.g. video recordings or computer screen experiments). By this means, we extended McNeill's 2D depiction of the gesture space to *gesture space volumes* expanding in time and space. Similarly, we created *attention volumes*, a new way to visualize the distribution of attention in 3D environments.

Contents

List of Figures

List of Tables

List of Acronyms

DG	Description Giver 69
DRIVE	D*eictic* R*eference* I*n* V*irtual* E*nvironments* (see Chapter A) .. II
G trial	Gesture-only trial 70
GFP	Gaze-Finger Pointing 47
HCI	Human-Computer Interaction 35
IADE	I*nteractive* A*ugmented* D*ata* E*xplorer* (see Section 4.6) II
IFP	Index-Finger Pointing 47
OI	Object Identifier 69
PSOM	Parameterized Self-Organizing Map 58
S+G trial	Speech and gesture trial 70

Acknowledgement

The vision of computer systems understanding humans has fascinated me since my first serious interest in computers. Finally, in 2004, it was Gert Rickheit who gave me the support and some freedom to follow my own research on multimodal deixis during my employment (2003 - 2005) in the Collaborative Research Center 360 (CRC 360), contiguous to my assignment to investigate verbal ellipses. He also helped me to find interesting peer groups of researchers at Bielefeld University, which laid grounds for this doctoral project.

I am especially thankful to Ipke Wachsmuth and Marc Latoschik, who believed in me when the first funding was over and gave me continuous support since then. Along my tasks in the EU project PASION (Psychologically Augmented Social Interaction Over Networks, 2006 - 2009) on social network analysis, first as a researcher, later as deputy project leader, I was therefore able to continue my research on multimodal deixis. In 2010, I finally concluded this thesis with the support of Ipke Wachsmuth, who employed me as research assistent in his group since the end of the PASION project in 2009.

In the field of multimodal deixis, my direct peers were Alfred Kranstedt, Andy Lücking, Hannes Rieser and Ipke Wachsmuth. We shared inspiring years in the CRC 360 and some of the cooperations continued fruitfully since then. Thank you very much! My special thanks go to Hannes Rieser and Ipke Wachsmuth for their cooperation and support throughout these years.

My research on interpreting ellipses in speech brought me into contact with Petra Weiß, Constanze Vorwerg and the eye-tracking group of the CRC 360, namely Kai Essig, Sven Pohl and Lorenz (Max) Sichelschmidt. This is when my interest in eye movements started and the idea arose to track the point of regard in space. Finally, when starting my doctoral project, I already had a strong peer group in the area of human-computer interaction, especially Timo Sowa, Stefan Kopp, Marc Latoschik, Christian Fröhlich, Peter Biermann and Ipke Wachsmuth. Thank you very much to all of you for your input and for all the discussions. Special thanks go to the student workers for their

support, especially Nikita Mattar and Dennis Wiebusch, who helped me in the programming and the conducting of the study on gaze tracking in immersive virtual reality.

I will forever be indebted to my family for their support. Special thanks go to my wife Nadine, who supported me more than she will ever be aware of. Thank you very much! Lucy, Miro, I have been working on my doctoral thesis your whole life long. This is going to change, big promise!

Chapter 1

Introduction

When humans communicate, we use deictic expressions to refer to objects in our surrounding and put them in the context of our actions. In face-to-face interaction, we can complement verbal expressions with gestures and, hence, we do not need to be too precise in our verbal protocols. Our interlocutors hear our speaking; see our gestures and they even read our eyes. They interpret our deictic expressions, try to identify the referents and – normally – they will understand.

From its beginnings, the driving vision of informatics has been to create computer systems which are capable of understanding and fulfilling our needs without us having to learn dedicated user interfaces. Current research following this vision can be found in attentive systems and situated communicative systems, where the human's way of interacting governs interface design. Systems should adapt to humans, not the other way round. Research in this area is thus highly interdisciplinary and disciplines such as psychology, linguistics or neurobiology provide relevant information to guide the design of such natural human-computer interfaces.

The overarching aim behind this thesis are multimodal conversational interfaces where humans are engaged in natural dialogues with computer systems. The embodied conversational agent Max developed in the A.I. group at Bielefeld University is an example of such an interface. Max is already able to produce multimodal deictic expressions using speech, gaze and gestures, but his capabilities to understand humans are not on par. If he was able to resolve multimodal deictic expressions at the same level as he is producing them, his understanding of humans would increase and interacting with him would become more natural.

1.1 Motivation

The primary questions addressed by this thesis are elaborated in this section based on a reflection of deictic gestures in the context of two models for communication – one from linguistics and one from mathematics/informatics. A technical account of pointing gestures is presented based on Communication Theory (Shannon, 1948), which addresses the *accuracy* of the transmission of a deictic reference, but neglects the content of the message that is exchanged. This approach is complemented by a semiotic approach (Peirce, 1965), which concentrates on the *meaning* of the deictic reference.

1.1.1 Communication Theory

The general communication system described by Shannon (1948) in his mathematical model sets the stage for human communication: an information source communicates messages to a destination (see Figure 1.1, left). A direct transfer of the messages is not possible, as sender and destination are separated by matter. The information source thus makes use of transmitters to convert messages to signals that can be transmitted over channels through the matter towards the destination. The destination operates one or more receivers, appropriate for the channels, and reconstructs the messages.

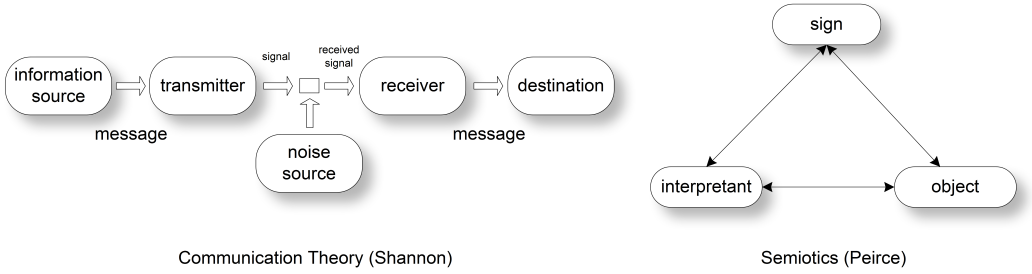


Figure 1.1: *The schematic diagram of a communication system from Shannon (1948) to the left describes the technical aspects of communication. The accuracy of the direction of pointing is one such aspect. Semiotics is concerned with the meaning of signals (diagram of Peirce's triangle of meaning to the right) and involves the question of which objects the interlocutor will refer to using a pointing gesture.*

In natural face-to-face interactions, we, as information sources or *senders*, use different transmitters, or *modalities*, to communicate. We produce signals using speech, facial expressions, gestures or eye gazes and distribute them over different channels (visual, aural). The focus of Shannon's model is on the

accuracy of the transmissions and not on the semantics of the message which has been transferred. In fact, he judges the semantics as being irrelevant to the engineering problems he is adressing.

The message of a pointing gesture, in the sense of Shannon's model, is defined by the *point in time* when the pointing gesture is expressed and the *direction* the pointing gesture is pointing to. A pointing gesture has been successfully transmitted whenever the receiver decodes the time and the direction of the gesture with a certain accuracy. The correct decoding of this "pointing message" encoded in the interlocutor's pointing gesture is the first challenge for a human-computer interface. This thesis addresses this challenge and creates models for the direction of gaze and manual pointing. These models stipulate the relevant parameters which need to be identified to accurately decode the direction of pointing gestures. As a consequence, initial questions of this thesis are:

- When does the interlocutor perform a pointing gesture, and what is the relevant time interval of the whole gesture trajectory?
- Where does the interlocutor point to (direction)?

1.1.2 Semiotics

Once the "pointing message" has been decoded and the more technical part of understanding pointing gestures is solved, Shannon's model should be complemented by a model from the field of semiotics, which is concerned with the meaning of signs. Peirce (1965) distinguishes three types of signs, among them the *index*, a sign with a direct connection to the object it refers to (see Figure 1.1, right). The pointing gesture is the sign and the task of the receiver is to identify the object the interlocutor is referring to with the gesture. As a consequence, a central question of this thesis is:

- Which object does the interlocutor refer to with the pointing gesture?

In this thesis, models for the extension of pointing gestures are developed which can be used to identify referent objects.

An aspect which cannot be found in Shannon's model, but which is explicit in semiotics, is that the recipient is free to decode any of the signals the interlocutor is transmitting, whether they are given intentionally, unintentionally or even involuntarily. Eye gaze is an example for a signal which can be used intentionally, to explicitly refer to something, or unintentionally, when attending to the object during the formulation of the verbal expression. The

recipient can decode the eye gaze of the sender in both cases and identify the object being referred to. Hence, in this thesis, the term *gaze pointing* refers to explicit and implicit *interpretations* of gaze which is referring to objects.

Summing up, two types of model are developed and tested in this thesis:

- models for the *direction* of pointing gestures and
- models for the *extension* of pointing gestures.

An accurate model for the direction of pointing gestures is an essential component of the model for the extension of pointing gestures. In this thesis, the term *pointing model* will be used to refer to the model for the extension of pointing gestures (which includes the model for the direction). An explicit reference to models for the direction of pointing is provided if necessary.

1.1.3 Research Context at Bielefeld University

The environment provided by the A.I. Group of the Faculty of Technology at Bielefeld University has provided an excellent atmosphere for this dissertation project, both on the personal and the technological level. The group's two decades of research in knowledge-based human-machine interfaces, virtual agents and interaction technology for virtual environments has built up extensive knowledge in this domain and has further provided an exhaustive set of tools and equipment that facilitated this research.

Communication between humans and between humans and machines has been the research focus of two Collaborative Research Centres (CRC) established at Bielefeld University by the German Research Foundation (DFG). The first CRC 360 (running from 1993 to 2005), *Situated Artificial Communicators* (Rickheit & Wachsmuth, 1996), motivated and funded the work on manual pointing gestures presented in this thesis. The subsequent CRC 673, *Alignment in Communication* (Rickheit & Wachsmuth, 2008), started in 2006 with the aim to investigate the more subtle, resource-conserving processes enabling human communication. To this ends, the research on manual pointing gestures started in the frame of the CRC 360 has been extended to include gaze pointing gestures in the CRC 673.

1.2 Thesis Scope and Objectives

The main motivation for this thesis is the improvement of conversational interfaces by empowering them to understand natural deictic expressions using gaze and manual gestures. Consequently, the main focus of this thesis is on studying human-human interaction to derive accurate models of human manual pointing gestures, which ultimately inform the conversational interface on how to understand human deixis. In addition, a second focus is on the advancement of gaze interaction technology, as human eye gaze conveys more often references to objects than manual pointing gestures do. Accordingly, this thesis is embedded in a highly interdisciplinary area of research. The original contribution of this thesis is a thorough investigation of the construction of reference from a detected pointing gesture.

Following this research program, the following challenges have been identified:

1. Accurate models for human pointing have to be found.
2. Precise data on multimodal interactions has to be collected in order to create these models. This data is multimodal (transcripts, voice and video recordings, annotations) and not directly accessible for analysis (voice and video recordings).
3. Technologies have to be developed to support the collection and the analysis of the multimodal data.
4. The created models have to be implemented, evaluated and optimized until they allow a natural interaction with the conversational interface.

The four cornerstones of the contribution of this thesis are detailed in the following sections.

Deeper Understanding of Human Pointing This thesis aims at substantiating our knowledge on human pointing. We want to find answers on the *when*, *where* and *which* of pointing gestures; questions which have been stated in Section 1.1. For this, we will construct sound models of human gaze and manual pointing which can be used as a basis for the development of conversational interfaces. As a consequence of this requirement, the models will be formalized in terms of mathematical expressions.

Advancement of Gaze Interaction Technology Gaze plays an important role in human communication, not only in deictic reference. Yet, tech-

nology for gaze interaction is still less developed than technology for manual interaction. To allow the user to move around freely in our envisioned multimodal conversational interface, we will develop innovative tools for the real-time tracking and integration of eye movements with body movements. This includes an estimation of the direction of gaze into 3D space. An even greater achievement is the detection of the location of gaze in 3D space.

Improvement of Conversational Interfaces To be applicable in human-computer interactions, the developed models of gaze and manual pointing have to be integrated into a framework for multimodal deixis in virtual environments. As a consequence, the framework has to support several levels of abstraction:

- integration of sensor devices, such as eye tracking systems or motion capturing systems
- extraction of relevant features from the raw sensor data
- detection of pointing gestures based on the extracted features
- interpretation of pointing gestures based on the spatial context

A framework constructed this way should provide viable information which can be used to implement higher processes of communication such as turn-taking, check-backs, shared attention and, finally, the resolution of deictic references.

Cross-Cutting Issue: Advancement of Scientific Methods The thorough investigations of pointing gestures undertaken in this thesis call for new methodologies for research on multimodal deixis in the fields of linguistics and computer science. The movements of the gestures have to be recorded in 3D to ensure highly accurate data which is free from perspective distortions – a problem which is often encountered when using 2D video recordings. This is crucial for the correct identification of the direction of a pointing gesture. At the same time, there exists little prior knowledge on how to preprocess and visualize the recorded 3D data, especially if aggregating over several participants. Several other sources of data have to be integrated as well, such as audio and video recordings or manual annotations. This thesis thus has to develop techniques to make use of the recorded data in an appropriate way.

1.3 Thesis Structure

Human pointing with hand and gaze has been observed and modeled by several scientific disciplines and thus information valuable for the design of a human-computer interface are summarized in an *interdisciplinary review* in Chapter 2. More technical aspects and innovative approaches demonstrating the use of gaze and gesture in the *human-computer interface* are discussed in Chapter 3. At the end of Chapter 3, a classification of different approaches to model the direction and the extension of pointing gestures is given, laying the basis for the following empirical studies and the modeling of pointing gestures.

Chapter 4 addresses *manual pointing gestures*. This chapter describes an extensive study which has been conducted as a joint effort of computer scientists and linguists to create a multimodal corpus on manual pointing gestures in a human-human dialog game. In this context, the chapter also presents the innovative tool IADE which has been created to record, analyze, simulate and explore the data recorded in the study. Unusually for a linguistic study, the recordings were made in 3D using motion capturing. For the analysis of gestures in 3D, *Gesture Space Volumes* have been developed which provide a coherent picture of the temporal and spatial development of gestures. This chapter presents important findings on the interaction between speech and gesture and on commonly occurring pointing strategies. It also provides first insights into *where* manual pointing gestures are targeted to.

Gaze pointing is investigated in two studies presented in Chapter 5. A first study on *direction-based gaze pointing* evaluates the accuracy achieved in detecting gaze pointing of a free moving user with a combination of gaze and body tracking. A second study evaluates different techniques and devices for locating the *point of regard in 3D* for *location-based gaze pointing*. This chapter also presents visualizations of the 3D gaze scanpath to support the analysis and assessment of the obtained results. This chapter offers first insights to the *where* of gaze pointing.

Chapter 6 focuses on the development of *models for the direction and the extension of pointing* with a main emphasis on models for manual pointing. To this end, this chapter reconsiders the corpus presented in Chapter 4 and proceeds step-by-step in a *data-driven modeling* approach. The resulting models are evaluated and visualized using the *Gesture Space Volumes* described in Chapter 4. Subsequently, the chapter considers gaze pointing and provides a refined model for the extension of gaze pointing. The results of the study on location-based gaze pointing are processed into *Attention Volumes*, newly

developed visualizations for the distribution of visual attention. This chapter gives answers to the *where* and *which* of manual and gaze pointing.

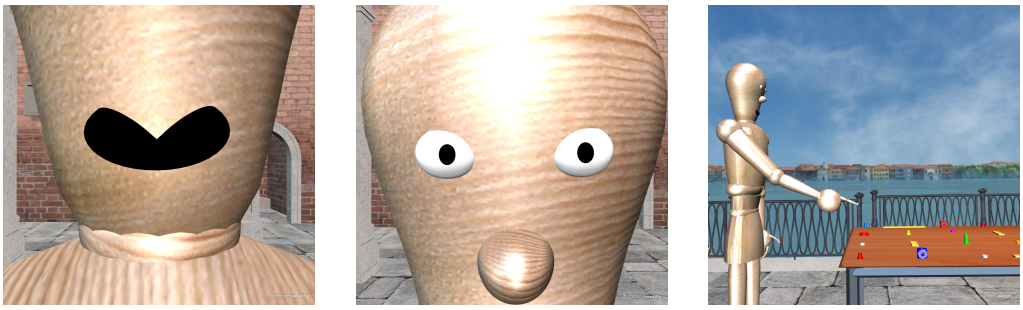
Chapter 7 presents applications to demonstrate the use of the developed models and technologies in different contexts. This chapter finally concludes the main part of this thesis with a résumé highlighting the findings and technical advancements achieved in this thesis.

Appended to this thesis is a description of the DRIVE framework for *D*eictic *R*eference *I*n *V*irtual *E*nvironments. DRIVE has been developed in this thesis project to implement the models on human pointing (Appendix A).

Chapter 2

Interdisciplinary Background

When confronted with the topic of *deictic expressions*, most people might think of speech and gestures, most specifically manual pointing gestures (see Figure 2.1 a and c). The role of the human eyes, besides their being necessary to perceive the interlocutors' pointing gestures, might be less obvious (see Figure 2.1 b). Whilst senders use explicit eye gaze pointing less frequently than manual pointing, the attention of interlocutors is strongly drawn toward the eyes of their interaction partners, and the direction of the other's attention. Due to the dual function as sensor and actuator, relevant eye gaze is much more difficult to detect than manual pointing gestures. Humans are experts at interpreting the involuntary sensory movements of their interlocutors' eyes, as the following review will show. From the recipient's perspective, the role of gaze might be equally important for interpreting deixis as that of manual gestures.



(a) Speech (b) Gaze (c) Gesture

Figure 2.1: *Speech, gaze, and gesture contribute to human communication.*

The first section of this chapter provides an overview of the use of gaze and manual gesture in communication. The following section discusses the concepts of reference and deixis from a linguistic perspective and provides a formal foundation for these terms. The next two sections concentrate on the individual contributions of manual gesture and gaze to deictic reference, with a special focus on how to derive the direction toward and, if possible, the location of objects referred to using these modalities.

2.1 Gaze and Manual Gesture in Communication

Research on human communication tends to be dominated by verbal communication. This is *inter alia* expressed in the common distinction between verbal and non-verbal communication. It is interesting to note that most language production and reception models in psychology used to concentrate on verbal communication only. One reason could be that speech-symbolic processing is quite accessible to computers. In contrast, simulating non-verbal behavior requires an embodiment of the communicator with human-like features. This view is shared by Herrmann (1982, pp 6f), who attributed the imbalance in the discussion of verbal and non-verbal communication mainly to the dominance of the *computer metaphor* in psychology.

The reference to the computer metaphor highlights how progress in basic research is sometimes tied to advances in technology and to a general availability of methods and tools. Similarly, research on verbal communication has primarily focused on the production side. The *how-to express* can be easily observed and content can be easily acquired. Findings can also be easily expressed and archived, as the object of investigation is naturally expressed using language and symbols. Research on non-verbal behavior on the other hand has primarily focused on the recipient side, the *how-to evaluate*, for similar reasons.

The advent of video technology introduced new possibilities for recording and – albeit with some restrictions – for reproducing non-verbal behavior. Today, body movements can be recorded in space and time with very high resolution using computer-based tracking systems. Other technologies, such as brain-imaging or eye tracking, provide us with more insight into intrapersonal processes. Finally, highly realistic artificial communicators can be produced using virtual reality technology or robotics to test multimodal models of

communication and to produce realistic, interactive stimuli for experiments in a controlled manner.

2.1.1 Terms

The following section reviews research from several disciplines. Before that, some terms are clarified which might otherwise lead to confusion between disciplines.

When talking about communication, computer scientists are accustomed to refer to the model introduced by Shannon (1948). In this thesis, the term *non-verbal modality*, which is more commonly used in the context of human communication, is used for the non-verbal communication link, instead of the more general term *channel*. Similarly, as the participants in face-to-face communication produce and process signals in parallel all the time, no strict differentiation between the roles of *sender* and *recipient* is maintained, but the more general term *interlocutor* is used instead. If appropriate, reference will occasionally be made to the *producer* of a pointing gesture or an utterance.

2.2 Reference and Deixis

In linguistics, the term *reference* can be found both in semantics and pragmatics (see Lyons (1968) for an overview). It refers to the relation between an expression, for example a noun, and the entities that are named by such an expression. First, there is the potential meaning tied to the expression (semantics) and second, there is at least one entity that is linked to such a referential expression, the *referent* (pragmatics). Intuitively, the referent of an expression depends on the context. The term *extension* refers to the set of possible referents of an expression, given a specific context (see Figure 2.2). Analogously, *denotation* refers to the constant meaning of an expression in semantics.

Deixis subsumes referential expressions used to locate and identify concrete or abstract entities within a certain context. Such entities could be people, objects, places, but also events or processes. The context comprises space and time (where/when is the expression produced?), as well as the interlocutors between whom the expression is exchanged. These expressions are called *deictic expressions*.

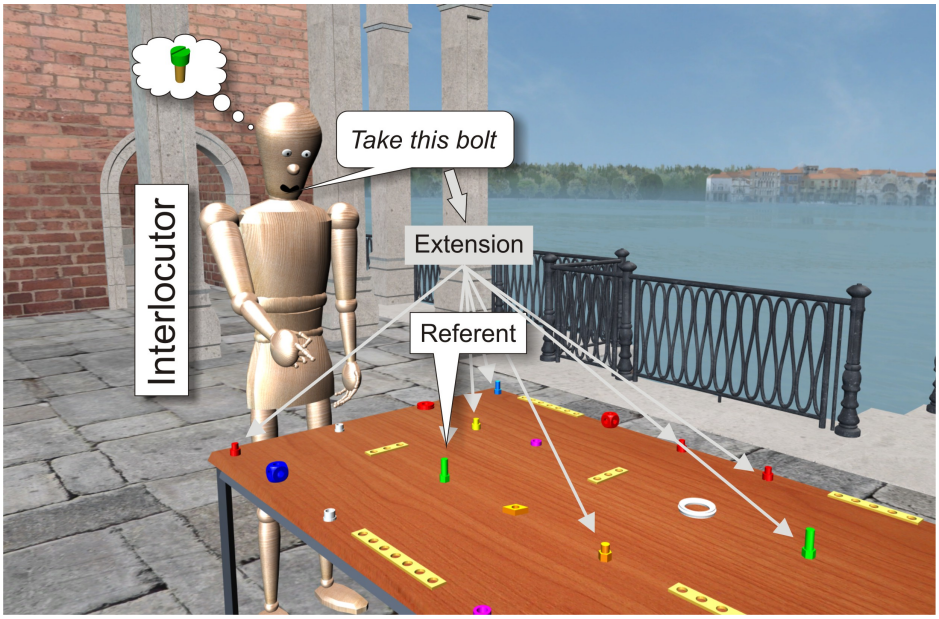


Figure 2.2: *Deictic expressions are used to refer to objects in the world. In the example depicted above, the interlocutor makes a deictic expression as part of a command. The intended referent object is the bolt with the green cap. The potential extension of the deictic expression in the speech alone covers a set of possible referent objects. The manual pointing gesture adds the required information to further restrict the potential extension to the intended object, the referent of the multimodal deictic expression.*

A successful exchange of deictic expressions depends on

- the properties of the producer of the expression,
- the quality of the medium (noise, etc.)
- and on the sensory abilities of the interlocutors.

Yet successful reference also depends on the interpretation of the expression by the interlocutors.

2.2.1 Deixis

Different categories of deixis can be identified (as discussed, for example, by Fillmore (1975) and Lyons (1977)):

place deixis : references to locations of entities. Examples are "here" and "there".

time deixis : references to time. Examples are "now", "tomorrow", but also the tenses used in an utterance.

person deixis : references to persons. Possible targets are the speaker ("I"), the addressee ("you"), overhearers, and third parties whom the utterance is about.

social deixis : references to social status. Examples are the German "Du" and "Sie" when referring to the addressee, which express different states of familiarity.

discourse deixis : references to parts of the ongoing discourse.

In addition, one can distinguish *symbolic* and *gestural usages* of deixis. If general knowledge is sufficient to establish the reference, it is called *symbolic usage*. If an active sensory process is needed to understand the deictic expression, it is called *gestural usage*. A typical case are deictic expressions that comprise a pointing gesture using the index finger, accompanied by a verbal expression like "this X". Besides such pointing gestures, the direction of gaze may also be part of a gestural usage of deixis. According to Bühler (1934), place deixis is one of the most basic means of human communication. It establishes the link between internal symbols and the entities in the exterior world.

This thesis concentrates on gestural usage of place deixis, in particular on pointing gestures using index finger and eye gaze. Pointing gestures performed by the arm, hand and index finger will be referred to as *manual pointing*. Pointing gestures performed with the eyes will be referred to as *gaze pointing*. Both types of pointing gesture will be investigated in detail in the two sections to follow.

2.3 Manual Pointing

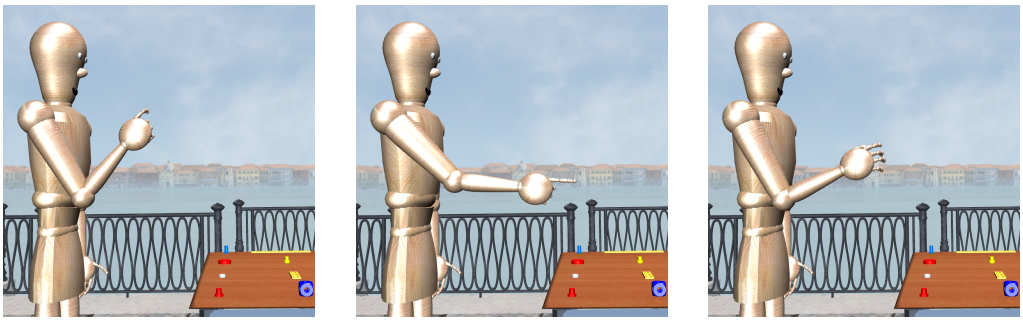
> *In pointing, the index finger and arm are extended in the direction of the interesting object, whereas the remaining fingers are curled under the hand, with the thumb held down and to the side* (Butterworth, 2003, p. 9).

Figure 2.3: *A manual pointing gesture modeled according to the quotation from Butterworth (2003).*

Butterworth (2003) provides us with a nice description of the prototypical manual pointing gesture, which is visualized in Figure 2.3. Note that in the quotation above, as in the literature on pointing, often a *totum pro parte* use of the noun "pointing" can be found when referring to manual pointing. This can be attributed to the strong tendency humans have to identify the extended index finger with pointing and vice versa. This convention will be adhered to in this section, where pointing in general and manual pointing in particular are discussed. The following section will thus make use of the introduced concepts and show how they apply to gaze pointing.

A comprehensive overview of the manifold gesture classifications in the literature has been assembled by Kendon (2004). According to Kendon, most authors recognize the special role of pointing gestures and create a unique category for such demonstrative gestures. In addition, McNeill (1992) distinguishes between concrete pointing, i.e., pointing at objects and events in the concrete world, and abstract pointing, i.e., pointing at abstract concepts. This thesis concentrates on concrete pointing, but the definition of concrete pointing needs to be modified for use in virtual reality: this thesis concentrates on pointing gestures targeted at entities which can be perceived in the current situation of the pointer and do not require that the entities themselves have a physical manifestation in the world (otherwise virtual and mixed reality applications would be excluded).

A great deal of movement may be involved in a manual pointing gesture. The climax or apex of the gesture with the index finger in the intended position takes only a fraction of the overall time of the gesture. It is thus useful to structure gesture movements in order to pin down the very moment where the



(a) Preparation (b) Stroke (c) Retraction

Figure 2.4: *The main phases of a manual pointing gesture as defined by Kendon (1980).*

intended direction is delivered. Kendon (1980) introduced a classification that has been widely adapted and extended (see e.g. McNeill, 1992). He identifies three main phases (see also Figure 2.4):

- *preparation*: from resting/previous position to intended position
- *stroke*: the apex of the gesture at the intended position
- *retraction*: from the intended position back to a resting position

Right before and after the stroke, the movement can decline into *holds*, which, according to McNeill (1992), were later differentiated by Kita (1990) into *pre-stroke* and *post-stroke holds*.

2.3.1 Direction

Although it is not inherently evident whether a pointing gesture indicates a direction, a target location or a target object, the construction of a direction is the essence of the pointing gesture. An intuitive interpretation of the function of the pointing gesture is that of a vector directed by the producer to the referent (see also Figure 2.5):

> *Pointing seems a straightforward matter: You stick your finger out in the appropriate direction, perhaps saying some accompanying words, and your interlocutors follow the trajectory of your arrow-like digit to the intended referent* (Haviland, 2000, p. 14).

> *The prototypical pointing gesture is a communicative body movement that projects a vector from a body part. This vector indicates a certain direction, location, or object* (Kita, 2003, p. 1).
>
> *Pointing gestures are regarded as indicating an object, a location, or a direction, which is discovered by projecting a straight line from the furthest point of the body part that has been extended outward, into the space that extends beyond the speaker* (Kendon, 2004, p. 200).

Mathematically, a vector is defined by its direction and length. It can be applied to any point in space and thus has no specific origin. Deictic expressions, however, have a dedicated origin, or *origo* as Bühler (1934) calls it. Thus in addition to a direction and length, an origin is needed, which defines the starting point of the vector.

Commonly, the origin is defined by some point within the body of the producer of the pointing gesture. This could be, as Kendon (2004) seems to propose, the furthest point of the body part that has been extended. A plausible alternative for the origin could be the experienced position of the self of the producer, e.g. behind his eyes.

The quotation from Haviland (2000) provides a specification for the direction of the vector: it is given by the "arrow-like digit". The other authors contend that the direction is at least associated with a body part, but whether the body part itself or its movements provides the direction remains unclear.

If one is only interested in the direction of a deictic expression, the length of the vector can be neglected. For practical reasons it should then be normalized to one. These kind of vectors will be called *orientation vectors*.

McNeill envisioned a *gesture space* as a shallow disk in front of each interlocutor in which gestures are performed (McNeill, 1992). This view is also adopted here for now. Real pointing takes place in *physical space* and can be perceived by the interlocutors. In contrast, *immediate space* can be populated by abstract conceptual entities. As McNeill (1992) points out, these entities do not necessarily need to have a correspondence in physical space (pointing at the concrete vs. pointing at the abstract). Haviland (2000) consequently differentiates between several *gesture spaces*. His *local space* is the physical space (similar to McNeill's gesture space) and his *narrated space* is superimposed on the local space and contains the concepts pointed to. According to his view, a sequence of pointing gestures does not necessarily create a coherent gesture space. Goodwin (2000) later extended McNeill's gesture space to the interactive space constructed in multi-party dialog.

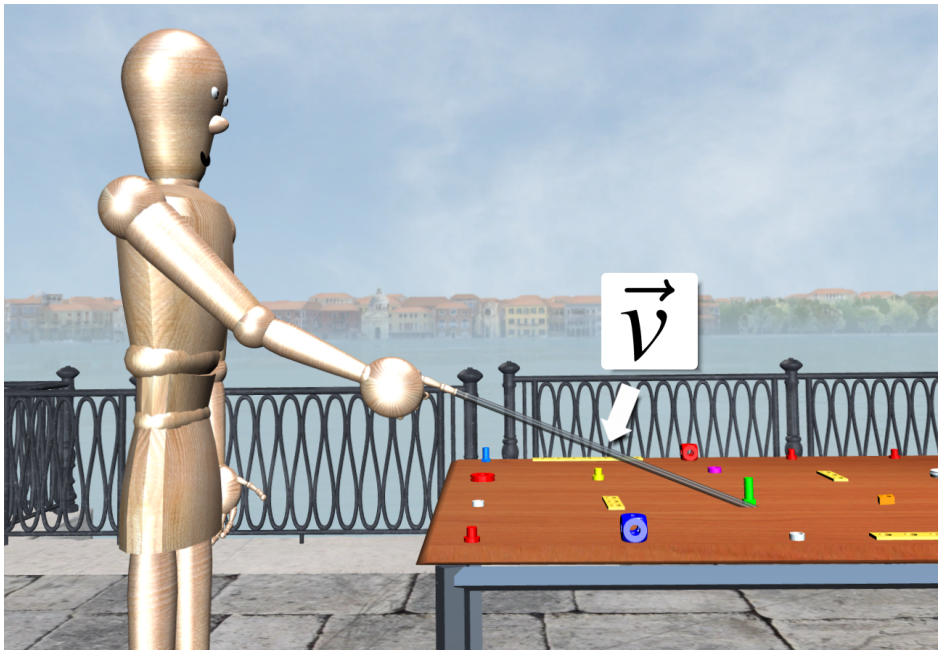


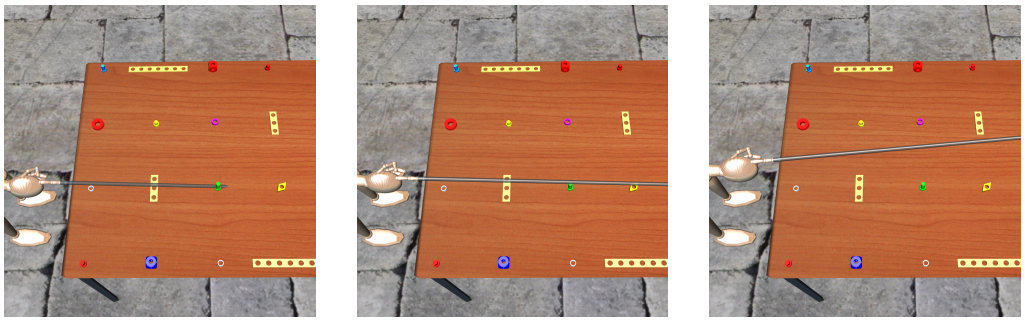
Figure 2.5: *The pointing gesture can be interpreted as a vector $\vec{v}$ directed at the referent.*

2.3.2 Location

Orientation vectors directed from the producer to the referent are not sufficient to specify a location. Thus, when the pointing is intended to refer to a location or an object, additional information is needed: the distance from the origin to the referent. This is a problem, because this information is not commonly found in pointing gestures using arms and hands. An exception may be when the producer of the gesture actually touches the referent object, so that the distance between origin and referent is zero.

If the pointing is intended to refer to an object, the line projected through the origin along the vector, the *pointing ray*, could be followed until it intersects with an object. The first object intersected by the pointing ray will then be the referent object (see Figure 2.6 a). A similar approach could be applied to locations, where the target surface is given by the ground, the walls, or some other physical manifestation of the location.

This approach is rather naïve and makes certain assumptions that do not easily hold, as can be seen in the study presented in Chapter 4. It requires that the interlocutors have already agreed upon which entities are relevant for the deictic expression. For example, a car could be a relevant entity in some



(a) Single object intersected

(b) Multiple objects intersected

(c) No objects intersected

Figure 2.6: *Determining the referent of a pointing gesture is problematic. Besides the ideal case (a), where the referent can, in principle, be identified, at least two other cases exist, (b) and (c), where this is not possible.*

dialogues as in: "Did you see my car?" Yet when talking to your mechanic it is more likely that the car is taken only as an aggregation of relevant entities, in order to locate the broken part that needs to be fixed. Another example consists of objects that are placed on the table, a situation that is used in one of the studies described in this thesis (see Figure 2.2). The interlocutors in that study were instructed to concentrate on the objects. In a different study, e.g. when talking about interior decoration, the table itself could be relevant, not the objects on top of it. Thus, rather than an arbitrary object that happens to be intersected first by the pointing ray, only *relevant* objects should be considered as potential referents.

In other situations, the pointing ray may intersect with several objects that are possibly relevant (see Figure 2.6 b). In this case the reference can only be established by a contextual inference. A similar situation arises if the pointing ray misses the referent object (see Figure 2.6 c). These examples are not uncommon and it can be questioned whether the pointing ray as such has a direct referential function.

Another problem consists of the requirement that the gesture be performed with a high accuracy, such that the constructed vector effectively hits the intended object. If this is not the case, the pointing gesture cannot establish reference at all. In addition, the interlocutors perceiving the pointing gesture must be very accurate in detecting the vector and extrapolating it to the target object. Given that the interlocutors may have very different perspectives, this seems unlikely when pointing at visually small objects.

2.3.3 Timing and Duration

Besides information about the spatial extension of a gesture, its timing is highly relevant for multimodal integration. McNeill (1985) found that the stroke phase of gestures coincided with speech in the majority of cases, later confirmed by Nobe (2000) after controversial discussion in the literature. Moreover, speech and gesture seem to form an integrated communication system (Kendon, 1980; McNeill, 1992; Mayberry & Jaques, 2000).

In their study on general gestures, Morrel-Samuels & Krauss (1992) found that the gesture onset preceded the onset of the lexical affiliate (the word in the verbal modality corresponding with the gesture) by between 0 s and 3.8 s (with a mean of 0.99 s and a median of 0.75 s). They also found that the asynchrony is correlated with the duration of the gesture: the longer it takes to perform the gesture, the earlier it starts. On average, gestures terminated 1.5 s (SD = 0.97 s) after their lexical affiliates.

With regard to manual pointing gestures and speech, Levelt, Richardson & La Heij (1985) present a series of studies on the timing of manual pointing gestures and speech. In their setting, they used a tracking system called Selspot to track the position of the index finger's tip with an active infrared LED in 3D. In the off-line condition (relaxed answer time) of their first experiment, they measured a mean advantage of 14 ms (SD = 100 ms) for the gestures' apexes (climax of the stroke) compared to voice onsets, whereas in the on-line condition (immediate answer), voice onset preceded apex by 53 ms (SD = 114 ms). These findings were replicated in a second experiment, where they investigated the difference in the timing of manual gestures with or without speech, among other things. They found that the apex onsets of manual pointing gestures accompanied by speech were delayed by 14 ms. While this difference in apex timings was not statistically significant, they did find a significant delay of the movement onset of the gesture, again by about 14 ms.

This thesis is primarily concerned with the *dwell time* of pointing gestures at maximum extension during the stroke. This is the time when the orientation vector is constructed. Ideally, the producer of the gesture remains motionless during this period, which is presumably not what is found in real situations. Therefore, small posture shifts should be allowed.

In a study on pointing to objects without direct feedback from the interlocutor, Müller-Tomfelde (2009) observed a median dwell time of about 1 s, with 50% of the data lying between 0.599 s and 1.598 s. He also presented the recorded

pointing gestures to interlocutors and found that pointing gestures were accepted after a median dwell time of 0.4325 s, with 50% of the data lying between 0.292 s and 0.552 s. Based on a third experiment, he finally concluded that feedback after dwell times of below 600 ms is experienced as natural.

Given these findings, it can be concluded that in co-verbal gestures the onsets of gestures usually precede the onsets of their lexical affiliates by less than a second. According to Levelt et al. (1985), manual pointing gestures are even more finely aligned with speech than gestures in general. Interlocutors producing manual pointing gestures use a dwell time of about one second if they are not given any direct feedback. Interlocutors observing manual pointing gestures accept them after a dwell time below 600 ms.

2.3.4 Accuracy of Gesture Recognition

Concerning the accuracy of gesture recognition, Butterworth & Itakura (2000) present a series of experiments in which they study the accuracy of gesture recognition for pointing with the combinations eye, head, head and eye, head and hand. The task for the test person was to identify one of three objects to which the instructor pointed. In their setting, the interlocutors were placed face-to-face but 5.95 m apart (see Figure 2.7). The target objects were located at a distance of 2.7 m from the wall where the test person was seated. The objects were positioned at angles of 5°, 15°, and 25° to each side of the test person.

They report that 4.5-year-old children could only manage to accurately identify the indicated location when the instructor pointed using head and hand. Even then, they could only accurately identify inner target objects or those in the right periphery. Head only was the next-best condition, followed by head and eye, and eye only, but all without statistically significant results. Overall, the availability of the experimenter's eye movements seemed to distract the children more than it helped. However, interlocutors seldom have their eyes closed while pointing in natural situations.

In an additional experiment, Butterworth & Itakura (2000) examined pointing recognition in adults in a similar setting (see Figure 2.8). This time both interlocutors were placed side by side. The position of the target objects meant that they appeared at angles of 4°, 6°, 8°, 10°, 15° and 45° from each other (only 4 objects in the latter case). In these studies, a limit of accurate spatial localization was at 15° separation between objects. For all pointing conditions, the test persons were able to identify the peripheral targets. In

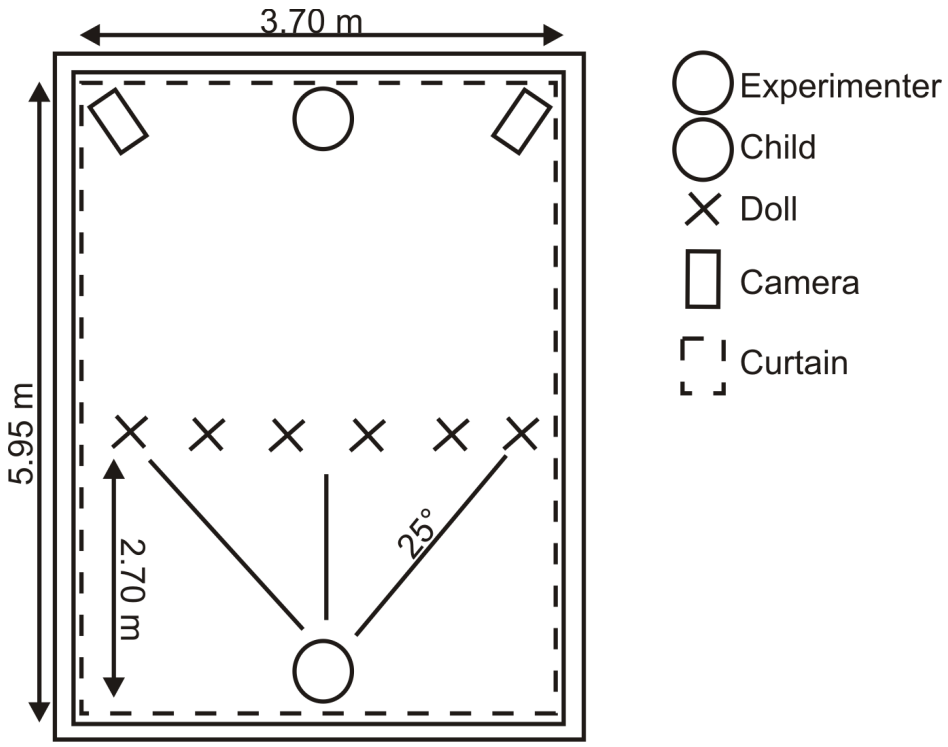


Figure 2.7: *Setting used in the study on pointing recognition in children by Butterworth & Itakura (2000) (redrawn from Butterworth & Itakura, 2000).*

the conditions without manual pointing, they could also identify the inner targets. Manual pointing actually obstructed the accurate localization of the inner targets.

At separations below 15°, peripheral objects were easier to identify for test persons than those at inner positions, except for the condition with eye movements only, where almost no correct identification was found for the peripheral objects.

Reviewing these results it can thus be concluded that in crowded areas where other potential targets are nearby, a minimum separation greater than 10° (they only tested for 15°) is needed to fully disambiguate pointing. Note that targets at the periphery do not follow this rule, which is an effect we also found in our own studies (see Section 4.9.1).

In the second study mentioned above by Butterworth & Itakura (2000), both interlocutors were sitting side by side and thus shared a similar perspective on the target objects. This positioning also implies that the test persons could have had perceptual problems with deictic expressions targeted at objects

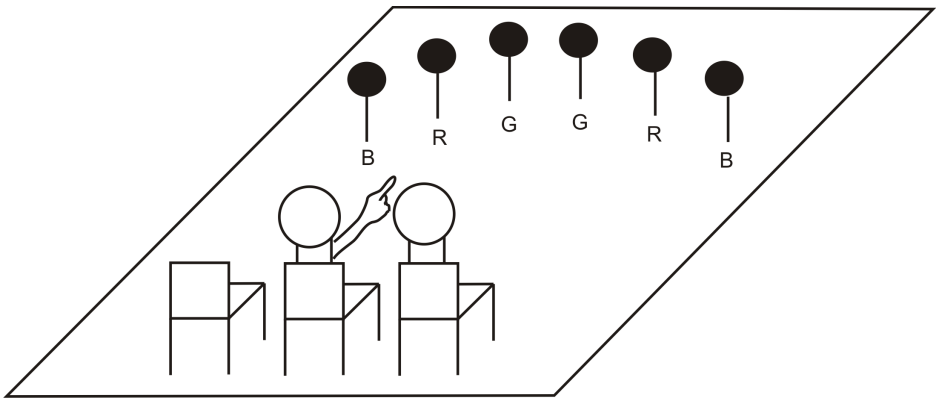


Figure 2.8: *Setting used in the studies on pointing recognition in adults by Butterworth & Itakura (2000) (original in Butterworth & Itakura, 2000).*

on the contralateral side. For example, it may not have been possible to see both eyes of the instructor except when demonstrating to the ipsilateral periphery. Unfortunately, this aspect is not discussed by the authors. In addition, nothing is said about the precision of the interlocutors' pointing gestures. It seems as if they were neither recorded nor were they exactly replicated between test persons.

In terms of the preceding study, Butterworth (2003) concludes that the precision of a vector-based interpretation of pointing is not sufficient to single out the referent and additional cues are required:

> *Thus if there is vector extrapolation it is at best approximate and sufficient only to differentiate between widely spaced, identical objects* (Butterworth & Itakura, 2000).

Butterworth & Itakura (2000) suggest that eye, head, nose and manual pointing may differ in the morphology of the space to which they refer, but they do not provide a proposal on how to model these spaces.

2.3.5 Visualization

Manual pointing gestures are swift body movements. Documenting them – and the specific aspects one is interested in – is not as straight forward as one might think. Individual pointing acts are often presented to an audience using short video sequences, but referent and referrer are rarely seen at the same time, except when pointing within personal space. Also, the perspective captured by the camera is static and may even be distorted. Photos or sketches

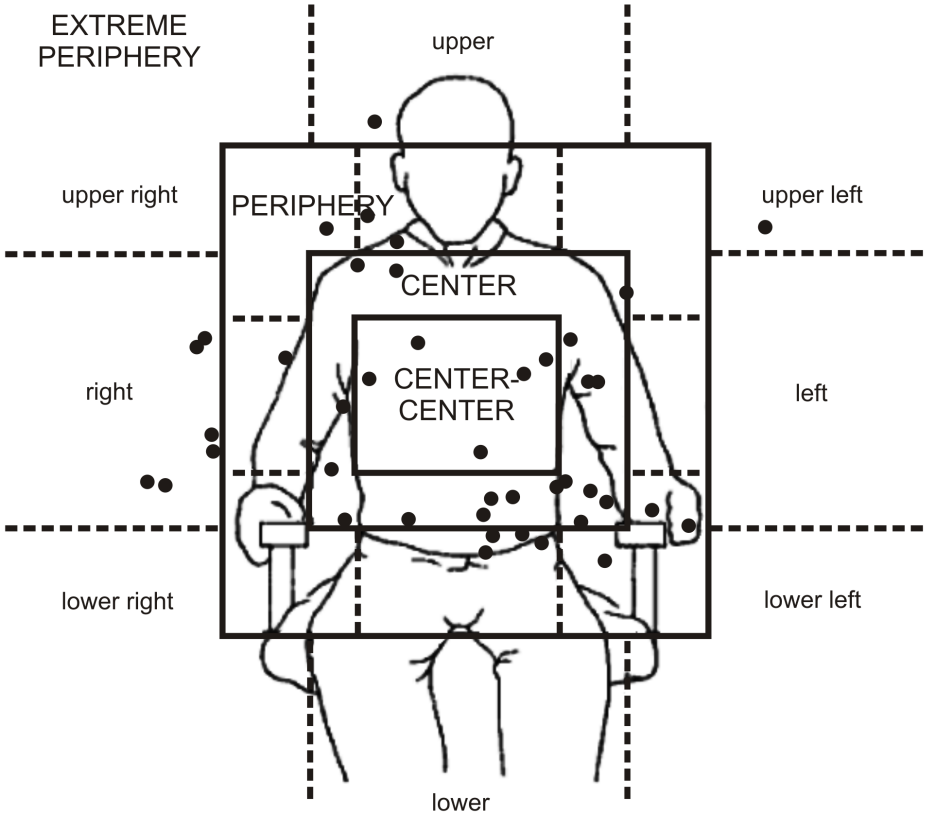


Figure 2.9: *McNeill's gesture space showing data of deictic gestures during a narrative (redrawn from McNeill, 1992, page 91).*

are also used, sometimes as a sequence, for example depicting preparation, stroke, and retraction phases. Sometimes these visualizations are enriched with arrows approximating the direction of the pointing.

The depiction of gesture space established by McNeill (1992) is one way to visualize multiple gestures. In his diagrams (see Figure 2.9), each point represents the position of the active hand during a gesture stroke, in this case within a deictic gesture.

In human-computer interaction, visualizations are provided as online feedback to the user, which forecasts consequences of ongoing interactions. Typical examples are pointing rays that shoot from the pointing device towards the object which would be selected if a certain action was triggered. Highlighting the target object is another example, as demonstrated by the Nintendo Wii-Remote: when pointing at the display, the position pointed at is marked

with a hand-like cursor (with extended index finger), the icons pointed at are visually highlighted and this effect is stressed even more by a rumbling feedback generated in the Wii-Remote controller itself. These visualizations document the essence of the pointing gesture, its direction and its extension in the application specific context. Yet they are not intended for documentation; they are transient and tailored to application specific needs. Summing up, gesture research lacks appropriate visualizations to document the dynamic movements of manual pointing gestures in 3D space.

2.4 Gaze Pointing

Butterworth & Itakura (2000) suggest that in ontogeny, comprehension of the line of gaze comes first, before comprehension of manual pointing. A major difference between manual pointing gestures and such usage of gaze is that manual pointing gestures are produced explicitly as part of a deictic expression and they are quite distinguishable. While there are occasions where gaze pointing is instituted explicitly, most commonly interlocutors will try to read their partner's mutual gaze direction regardless of their partner's intentions. This is due to the fact that humans tend to direct their visual attention to the objects they are talking about, or to be even more precise, to the objects they are going to talk about. Interlocutors have learned to follow the visual attention of their opposite (see Section 1.1.2 on semiotics).

In this thesis, the term *gaze pointing* is used metaphorically to refer to eye gaze directed at objects to express the commonalities between following the visual attention of interlocutors and following their manual pointing gestures when interpreting deictic expressions.

2.4.1 Visual Attention

Our eyes allow us to perceive the world visually. They are, however, optimized to see very accurately only within a small area of the retina, the *fovea centralis*. The area in which the highest acuity can be achieved covers only 2° of the visual field, the zone of high acuity extends up to 5° (Duchowski, 2007). The lines projected from the environment through the center of the eye on the retina are called *visual lines*. The *visual axis* of an eye is the visual line stimulating the center of the fovea centralis.

This implies that if an object is inspected, the orientation of the eyes is changed, so that the projection of the object onto the retina falls (partly) onto the fovea centralis to achieve high acuity. The eyes are thus rotated toward a specific direction and then rest on the desired object, more-or-less maintaining their orientation for a small period of time. These fast rotations (up to $1000°/sec$ are possible) are called *saccades* (20 ms to 200 ms, depending on amplitude) and the periods of rest are called *fixations*. It is during fixations, which last for about 150 ms to 600 ms (Duchowski, 2007), that humans obtain their visual input and attend to their surroundings visually.

Two different types of this visual attention have been identified: *covert visual attention* does not require shifts of the eye; *overt visual attention* is always accompanied by detectable movements of the eye. Mutual awareness of overt visual attention is an aspect of joint visual attention, as is the mutual awareness of manual pointing. In the following, this thesis will concentrate on the aspect of overt visual attention alone when referring to gaze.

2.4.2 Direction

In manual pointing, the direction has been modeled using a single vector, constructed by the index finger. In gaze pointing, the referent can be assumed to be the target of overt visual attention. During a fixation, the direction of overt visual attention can be derived from the visual axis of the pointing eye.

The visual axis does not necessarily need to be aligned with the referent object. It could be any visual line hitting the fovea centralis. Thus, instead of a single vector, one is faced with a series of vectors. An appropriate model for this is a cone with its apex in the eye, a central axis along the visual axis and an opening angle defined by the angle of the fovea centralis (2°).

Chi & Lin (1997) tested the horizontal and vertical accuracy of gaze pointing using an ASL-4000 eye tracker in a fixation task on a computer-screen (50 cm distance to user). In their setting, the user had to maintain a still head position which was ensured using a chin-rest. They found a difference in accuracy depending on the vertical position of the target. For targets presented above or coinciding with the horizontal line of gaze, their advise is to use visual targets with a width of 2.0° and a height of 2.4°, while for targets presented below the horizontal line of gaze, they recommend using visual targets with a width of 2.6° and a height of 3.9°. The results above the horizontal line of gaze nicely fit to the model of a cone with an opening angle of 2.0∘. Regarding the results for the lower part it is unclear, whether they are due to features

of the visual system, or due to constraints of the optical eye tracking system, which might have not a full view on the pupil when looking downwards (e.g. due to occlusions by the eyelid or eyelashes).

2.4.3 Location

While the identification of the direction of gaze introduces some uncertainty, there is an advantage of gaze pointing over manual pointing: overt visual attention of both eyes will generally be aligned on the same referent object. This provides two origins and thus two directions from slightly different perspectives which can be use to determine the location of overt visual attention, also called the *point of regard*.

Although the retina of the human eye only samples a two-dimensional projection of the surroundings, humans are capable of reconstructing a three-dimensional impression of our environment by adding depth. In the literature (Goldstein, 2002, e.g.) several criteria for the construction of depth perception can be found:

monocular depth criteria such as *occlusion*, *relative size/height in the field of view*, *common size of objects*, *atmospherical and linear perspective*, the *gradient of texture*, or *motion parallax* convey spatial information with a single eye only.

binocular depth criteria include *disparity* (differences in the retinal picture caused by the disparity of the eyes), *vergence* or *accommodation*.

Binocular depth perception, *stereopsis*, provides a way to differentiate the depth of objects up to a distance of about 135 m. In determining the depth of a fixation, only such criteria can be used that require measurable and thus perceivable effort from the perceptual system. As most of the criteria listed above do not have a sensory-motor component, only vergence and accommodation remain for consideration. Both vary depending on the distance of the fixated object.

If the projection of the object onto the retina falls (partly) onto the fovea centralis of both eyes, the images can be fused. Only then can one have a binocular fixation. Two categories of eye movements are distinguished: when the eyes follow an object moving horizontally or vertically in the same direction, they are called *version movements*, and when the eyes move locally in opposite directions, they are called *vergence movements*. Vergence movements are those associated with objects altering their depth. The horizontal component

of the movement is relevant for stereoscopic depth perception (Wheatstone, 1838). Measuring vergence angles, one may differentiate fixation depths up to a distance of 1.5 m to 3 m depending on the user's visual faculty. Thus, the working range of vergence movements nicely covers typical interaction spaces in face-to-face communication.

In regards to accommodation, a healthy eye of a young adult has an operational range between focal lengths from 1.68 cm to 1.80 cm. Thus differences in accommodation can theoretically be measured for distances between approximately 0.25 m and 100 m. At the time of writing, these measurements are only possible with research prototypes of vision based eye trackers (Suryakumar, Meyers, Irving & Bobier, 2007), but not with off-the-shelf technology. Thus, for the time being, only vergence movements can be used to measure the position of the point of regard in 3D space. However, at least to the author's knowledge, there is no evidence so far that the other's accomodation is used by interlocutors at all.

2.4.4 Timing and Duration

The timing of eye gaze is a highly relevant cue for its interpretation. The stroke of a manual pointing gesture can be identified as an apex of the gesture. There is no such apex in eye gaze, only fixations. Also, fixations are not used solely to explicitly refer to objects; they are mostly used for perception and search processes.

Fortunately, the different tasks behind eye movements can be identified by their different processing times. Velichkovsky et al. (1997) have compiled findings to create a diagram (see Figure 2.10) that shows the variation of fixation duration depending on the purpose of fixation. The diagram shows that short fixations (below 250 ms) can primarily be attributed to localization and figurative integration processes. The fixations associated with the processes of interest in this thesis, such as semantic or selfreferential processes and processes of communication, have durations of 250 ms or greater.

2.4.5 Reading the Eye: Gaze Awareness and Recognition Accuracy

Summing up: humans look at referent objects when they utter deictic expressions; and details have been presented on the speed, the accuracy and the

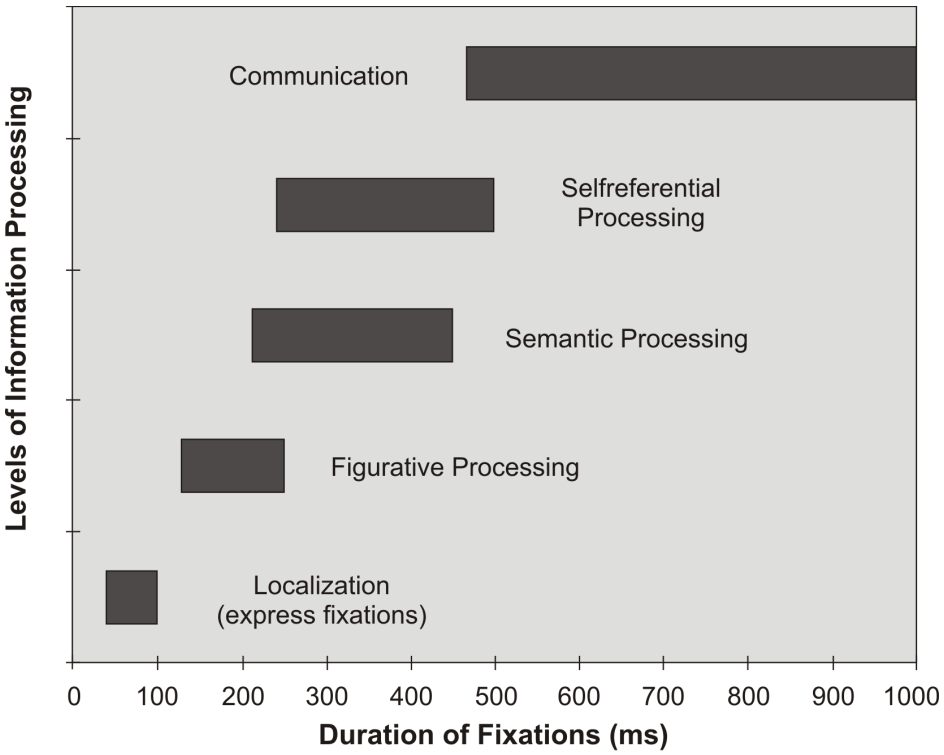


Figure 2.10: *Typical fixation durations for different tasks (redrawn from Velichkovsky et al., 1997).*

duration of these fixations. The question remains whether or not interlocutors are aware of these factors and make use of them?

Gale & Monk (2000) define three types of gaze awareness. Knowledge about the referent being looked at is coined *full gaze awareness*. In *partial gaze awareness*, only the general direction in which someone is looking is recognized. The third class is that of *mutual gaze* or eye contact, i.e., when interlocutors are looking in each other's eyes.

Mutual knowledge about full gaze awareness can be used as a conversational resource. In their experiments on video-mediated communication, Monk & Gale (2002) report a reduction to half the number of words and turns using localization tasks when providing full gaze awareness. Their findings emphasize the important role of full gaze awareness as a conversational resource in face-to-face dialog.

Gibson and Pick (1963) report an accuracy of 2.8° in recognizing the direction of mutual gaze. Cline (1967) reports errors of about 1.25° vertically and 0.75° horizontally. Gale & Monk (2000) investigated the accuracy of gaze in

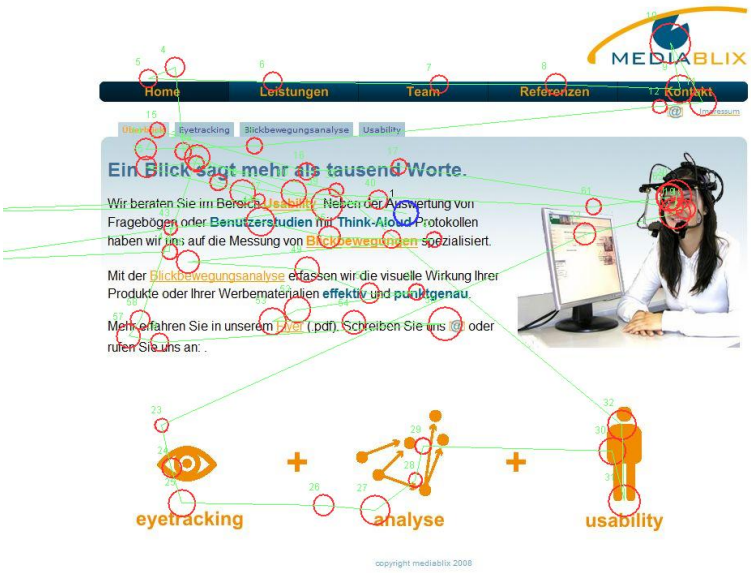


Figure 2.11: *The gaze path shows the sequence of fixations of a single user when visiting a web page. Fixations are represented as circles with a diameter correlating to the duration of the fixation. Saccades are represented as lines connecting fixations.*

full gaze awareness situations, i.e., when recognizing looks to referents. For combined head-eye-rotations they report an accuracy of 6° 84% of the time, and 12° 98% of the time.

2.4.6 Visualization

Information about eye movements on a specific visual scene is of interest for several research disciplines, for example, psycholinguistics or usability research. There are several ways in which to visualize results of an eye tracking session. For a single user, the gaze path, i.e., the sequence of fixations on a specific scene, the *scanpath*, is a common method (see Figure 2.11).

When data from several persons need to be integrated, heatmaps often prove useful to researchers (see Figure 2.12). Heatmaps use a color scale from cold (black, blue, green) to hot (yellow, orange, red) to depict the amount of attention the corresponding pixel has received over a series of trials (intra- and/or interpersonal). Heatmaps are overlaid transparently over the 2D stimulus to help establish the correspondence. These features make heatmaps

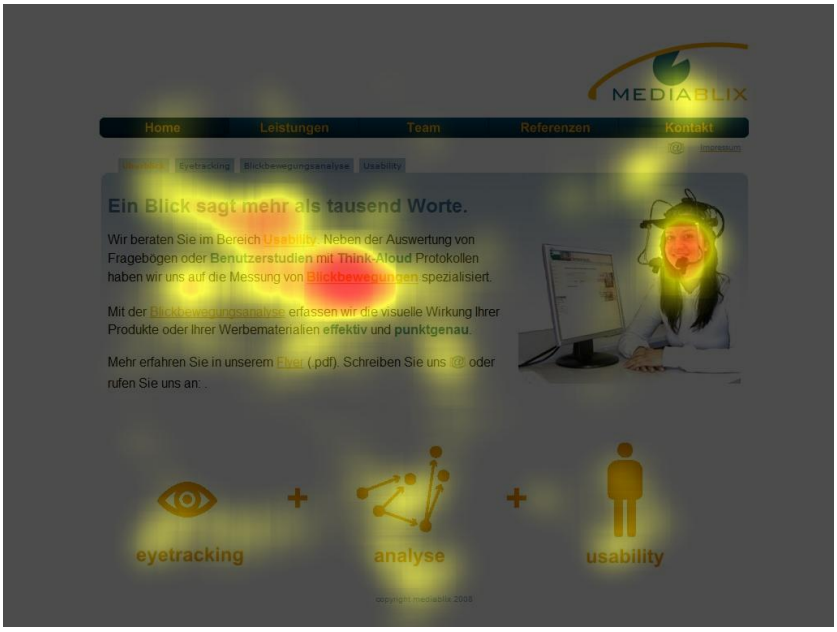


Figure 2.12: *The heatmap aggregates the visual attention from several users (here seven) and highlights frequently attended areas in red.*

easily accessible. The index visualized by the heatmaps could be any of the following:

- the count of fixations on a given pixel,
- the percentage of participants that attended to the pixel, or
- relative or absolute gaze duration on the pixel.

A discussion of the pros and cons of heatmaps can be found in Bojko (2009).

2.5 Coupling of Gesture and Gaze

Human manual pointing gestures are often preceded by a saccadic eye movement towards the target of the pointing gesture. In such situations, the speed of the eyes is so fast that they fixate the target before the hand movement is initiated (Bekkering et al. 1994, 1995; Frens and Erkelens 1991; Prablanc et al. 1979, 1986). The accuracy of pointing decreases when the visual target is not fixated (Abrams et al. 1990; Neggers and Bekkering 1999; Prablanc et al. 1979; Vercher et al. 1994). This is the case even if the target is no longer visible (Prablanc 1986).

Neggers & Bekkering (2000) show that eye movements and hand movements are interlinked. Eye movements also seem to be constrained to continue fixating the target until the hand has reached the target as well. In a follow-up study, Neggers & Bekkering (2001) show that this link is based on a non-visual, probably proprioceptive signal.

2.6 Summary

Pointing is used to link internal concepts with entities in the real world and thus contributes to a specific kind of reference. This chapter has distinguished between *manual pointing* and *gaze pointing*. Both are considered in linguistics under the concept *deixis*.

Conclusive findings have been presented addressing the *when* question of the timing of *manual gestures*. Kendon, McNeill, Kita and others adhere to a distinction between three main gesture phases, and provide detailed accounts on when the meaningful phase of the gesture is the stroke. If uttered co-verbally, manual pointing gestures and their lexical affiliates are tightly aligned, which simplifies the integration of gesture and speech in the multimodal interface. A dwell time between 300 ms and 600 ms has been found to be typical for a meaningful pointing gesture. These findings are similar to those for meaningful *gaze pointing*, where dwell times (duration of fixations) above 250 ms are interpreted as relevant, and a communicative function is attributed to those above 500 ms.

Concerning the *where* question, it has been argued that *manual pointing* is a prototypical deictic gesture and that there is a strong tendency to model its function with a vector oriented in the direction of the pointing index finger (or the most extended body part), but no precise account is given in the literature. This is different for *gaze pointing*, as a direkt link can be drawn between overt visual attention and pointing. If humans focus their visual attention on an object, they align the visual axis of their eyes with this object, with a minor angular deviation given by the area of high visual acuity, which is below 1° (given an opening angle of 2∘. It is thus reasonable to define the visual axis as an appropriate model for the direction of gaze pointing.

Regarding the *which*-question, a frequently used strategy is to identify as the one referent the object which is intersected by a vector extrapolated along the direction of the pointing gesture. However, empirical evidence suggests that this approach only provides dissatisfying results when inferring the referent

location or object of *manual pointing*. The extension of *gaze pointing* will typically be modeled by a cone describing the area of overt visual attention. In addition, the vergence movements of eye gaze have been identified as a promising source of information for deriving the depth of visual attention. Knowledge about the estimated distance of the referent object could further improve the success rate of pointing models, especially for cases of partial occlusions or to locally expand the area within which candidate objects are considered.

It is essential to develop versatile methods to visualize the data compiled in empirical studies on gaze and manual pointing. The traditional methods relying on 2D projections of the gesture space are not sufficient to accurately measure the direction of a pointing gesture. An additional challenge is the visualization of aggregated data, i.e. summarizing over several pointing acts or visualizing data of rapid eye movement sequences. Regarding eye gaze, heatmaps seem to provide valuable insights on the distribution of attention, but they are restricted to 2D content.

A review of the existing literature provided viable answers to the *when*-question for both gaze and manual pointing, and information about overt visual attention has provided a very accurate answer to the *where*-question of gaze pointing. Some questions, however, remain:

- *What exactly defines the direction of manual pointing?*
- *How accurate is the direction of manual pointing or gaze pointing?* Data on the recognition of pointing gestures has been discussed. Yet it remained unclear, whether the deviations can be attributed to the recipient alone or whether the producer of a pointing gesture also deviates from the ideal pointing direction.
- *How does pointing to objects work?* There is a tradition of associating pointing with a vector and somehow deriving the referent object based on this vector. Yet, to the knowledge of the author, no comprehensive model of this process has been proposed.
- *What is the shape of the referential space?* The vector model for pointing does not seem sufficient to describe the findings. Butterworth & Itakura (2000) suggest modeling different referential spaces for each pointing modality, but they do not offer any models.
- *Are there appropriate visualizations for 3D data on pointing gestures?* Existing methods concentrate on 2D contexts and are not sufficient to assess the direction of pointing in 3D space.

These questions are approached in this thesis using a *data-driven modeling* approach. To this ends, state-of-the-art tracking technology is used to record precise data on pointing, which among other things is discussed in the following chapter on the technological background of conversational interfaces.

Chapter 3

Related Work in Human-Computer Interaction

In Human-Computer Interaction (HCI), gestures are classically associated with mouse gestures or – less frequently – with pen gestures. Only recently, with the advent of multi-touch interfaces, natural finger and hand movements have entered the focus of the commercial mainstream. However, there has been continuous research, primarily in the lab, on HCI using natural human arm and hand gestures or eye gazes for controlling computer systems. This chapter provides an overview of the related work in this field, with an emphasis on systems that allow for the selection of objects via gaze or hand/arm gesture, and on those that combine these modalities with speech. A brief excursion into frameworks for multimodal integration and reference resolution in particular, provides insights into the general framework of conversational interfaces for natural dialogs.

3.1 Multimodal Interaction with Gesture and Gaze

Interaction techniques for 3D user interfaces can be assigned to three major categories: *selection*, *manipulation* and *navigation* (Bowman, Kruijff, Joseph J. LaViola & Poupyrev, 2005). The deictic expressions and pointing gestures under consideration in this thesis are techniques for object selection. The interpretation of deictic references is therefore, in terms of HCI, one aspect of 3D object selection. There are several other 3D object selection techniques

that are quite different from natural pointing. In the following only such techniques for 3D object selection will be considered that are similar or at least close to the deictic expressions used in human-human communication.

3.1.1 Starting Point

One of the first interactive systems supporting deictic expressions and manual pointing is Bolt's "Put that there" system (Bolt, 1980). In his work on a "Spatial Data-Management System" (SDMS), he developed a speech and gesture interface supporting direct commands to manipulate graphical items on a flat computer screen on a wall. The speech engine of his system, a DP-100 Connected Speech Recognition System developed by NEC America, Inc., recognized utterances of up to five words from a total of 120 different words in its active memory. The latency of the speech recognition was about 300 ms. The position and orientation of the hand was sensed using a magnetic field by a system from Polhemus Navigation Science, Inc. The developed interface was able to resolve verbal references using accompanied manual pointing gestures as in "Move that to the right of the green square", with "that" being the verbal affiliate of the manual pointing gesture. While Bolt's system required many pauses (in his own words: "the obligation to pause represents ... something of a breakdown in the general convenience of continuous vs. discrete speech input" (Bolt, 1980, 269)), it demonstrated the power of natural interfaces and inspired researchers all over the world.

A year later, Bolt (1981) envisioned a gaze interface ("eyes as input") to a multimedia installation "World of Windows" in a similar technical set-up. Based on the point-of-regard of the user, individual videos on a videowall were played and sound was mixed according to the history of videos attended to (so that when looking at a new video the sound of this video faded in while the sound of the video watched previously faded out). In contrast to the previous work on speech and manual pointing, the user did not explicitly command the system. Instead, the system was aware of the overt visual attention of the user and adapted its presentation accordingly.

In a follow-up paper, Bolt (1982) elaborated on these ideas:

> *We may note that the eye is a "pointer" par excellence. We can and do look at things in the visual field directly and steadily, micro movements of the eye ("tremor" and "drift") notwithstanding* (Bolt, 1982, 361).

This idea of gaze as a pointer is adopted in the work at hand. Nevertheless, Bolt's description of a "pointer" evokes the connotation of "mouse pointer" – a technical view that does not extend well to 3D space. In this thesis, gaze pointing is taken in a more natural sense, similar to manual pointing and not a pointing that precisely pinpoints objects.

3.1.2 Research on Gestures in the AG WBS at Bielefeld University

Wachsmuth, Lenzmann, Jörding, Jung, Latoschik & Fröhlich (1997) presented a system for interactive design and exploration with a speech and gesture interface. The system was able to recognize speech accompanied by pointing gestures recorded using a DataGlove and an Ascension Flock of Birds. A particularly interesting aspect of the system's interface was its personalization by a virtual character called Hamilton. Among other aspects, Hamilton made the frame of reference of the system explicit and, alternatively, could also be used as an avatar for the user, taking her or his perspective.

Subsequently, the group around Ipke Wachsmuth intensified their work on gestures, creating classification algorithms mapping manual gestures and hand shapes to a symbolic notation system in real-time (Fröhlich & Wachsmuth, 1998). This allowed for a top-level interpretation of the gestures, for example, by attributing meaning to a performed gesture using a gesture lexicon or by integrating gestures with speech (Sowa, Fröhlich & Latoschik, 1999). At the same time, Latoschik & Wachsmuth (1998) extended their work on pointing gestures to large screen installations and, in a follow-up paper, to direct manipulations, such as translating and rotating objects (Latoschik, Fröhlich, Jung & Wachsmuth, 1998). This work finally lead to ProSA (Latoschik, 2001), a framework for gesture detection, and the tATN (Latoschik, 2002), a framework for multimodal integration.

3.1.3 The Case of Gaze

Bolt's visionary papers (Bolt, 1980, 1981, 1982) anticipated many interesting advantages of gaze-based interfaces and provided reasonable starting points to approach the relevant challenges. Yet there is no follow-up paper elaborating on the success of his projects until Starker & Bolt (1990), who describe gaze-based interaction in a much simpler setting, with a user sitting directly in front of a small display screen. It must thus be assumed that for that

time Bolt's propositions were too ambitious, given the available hardware and software.

While Bolt was one of the first to provide a specific concept for the use of wearable eye tracking systems in human-computer interaction, the idea of using eye gaze to improve HCI had already been envisioned by Seymour Papert and Marvin Minsky in 1967 (Papert & Minsky, 1967). At the time they wrote their memo, eye movement analysis was done offline. Nevertheless, they anticipated great opportunities for HCI using online eye tracking. One of the examples they brought up is that of an information retrieval system where looking at an area representing a high-level concept triggers an expansion of this area to reveal next-level details. These ideas are quite similar to those realized in Bolt's work, and it is surely no coincidence that all of this work happened at MIT.

First practical uses of eye tracking in HCI concentrated on command interfaces for the disabled (Levine, 1981, 1984; Hutchinson, White Jr, Martin, Reichert & Frey, 1989). Yet more fascinating and also more challenging is the idea of using eye tracking for non-command interfaces (Nielsen, 1993), where the system should be enabled to infer the intentions of the user, for example by reading their eye movements. Jacob (1993) provides an early analysis and summarizes apparent difficulties regarding the interpretation of eye gaze, such as the Midas Touch problem (if involuntary and voluntary actions cannot be distinguished), natural jittery motions of the eye and shortcomings of the eye tracking equipment. At the end of his paper, he envisions a transition of window systems from the desktop to three dimensions (as realized in virtual reality) using these new styles of non-command interaction.

Ten years later, Jacob & Karn (2003) still attested to eye tracking in HCI as a promising approach. Nevertheless, after summarizing the state-of-the-art in eye gaze based HCI in 2003, they had to conclude that the slow rate of improvement in eye tracking equipment had thus far prevented a widespread adoption. Since then this has been in a continuous process of change, as the costs for the required hardware equipment are steadily decreasing while processing power and resolution are increasing at the same time (see Section 3.2.2 on page 48).

Pomplun, Prestin & Rieser (1998) used a monocular eye tracker to record gaze patterns during the inspection of a 3D toy airplane. The data was recorded on 2D video and analyzed offline to elicit data on planning and focus management. An interesting application where gaze was mapped onto objects in 3D space was developed by Rötting, Göbel & Springer (1999). They tried to improve the offline analysis of fixations on static real world objects in

2.5D, i.e. 2D pictures with a fixed perspective. Their system combined eye tracking with head tracking using a magnetic tracker. In a preprocessing step, the depth of relevant objects was determined. They manually marked the 2D bounding rectangles in at least two video frames taken by the scene camera attached to the head-mounted eye tracker. They then integrated the information about the movements of the regions in the local coordinate system of the scene camera with the position and orientation of the head. In this way they could derive a 2.5D reconstruction of the object position. However, this technique achieved only a coarse tube-shaped approximation of the object's 3D shape (rotating the bounding rectangle). Nevertheless, their technology had the potential to speed up post-hoc fixation analysis of real world eye tracking by a factor of six.

In the last decade, non-command interfaces have, for example, made use of the direction of gaze to improve visual fidelity at the point of regard. This can be done by rendering high-resolution geometry models in the center and models of reduced complexity in the periphery (Luebke, Hallen, Newfield & Watson, 2000; Murphy & Duchowski, 2001), by saccade-contingent updates (Triesch, Brian T. Sullivan, Hayhoe & Ballard, 2002) or, for example, by applying visual effects such as depth-of-field and camera motions (Hillaire, Lecuyer, Cozot & Casiez, 2008). These approaches adapt the visual display to the user's overt visual attention and may help to optimize the resource allocation of computing power, but they are not used to communicate with the user.

3.1.4 Multimodal Deixis for Conversational Interfaces

The type of interface developed in this thesis follows the collaborative manipulation metaphor (Hutchins, 1987): it is embedded in an environment that can be changed by the user using direct manipulation (Shneiderman, 1982). The users have an interface intermediary at their side, such as an embodied agent who functions as a conversational interface. The vision is that the user can communicate with the embodied agent as if it was a human, relying on natural language and gestures. This vision meets the criteria for Nielsen's noncommand interfaces (Nielsen, 1993), as the user is no longer required to interact with a tedious user interface, but can interact with the agent in a natural way, while the agent takes care of handling the system interface.

Embodied conversational agents (ECAs, Cassell, Sullivan, Prevost & Churchill, 2000) should be congruent in their capabilities of producing and understanding multimodal utterances. If one side is more developed, this could either not pay

off or even have a negative impact on the overall believability and performance. If, for example, the system is very good at understanding the user, but does not produce utterances on a competitive level, users might adapt to using simpler utterances as well, and the advanced system capabilities would lie idle. If the system produces very high-level utterances but fails to interpret similar utterances from the user, this might be even worse, as the user might get annoyed or frustrated.

The visual fidelity of virtual characters can already reach photo-realism and there are highly articulated robots in real life. Techniques such as motion-capturing or key-frame animation also allow for a realistic enlivenment of such agents. Yet, the real challenge is to produce natural looking gestures on-the-fly on a per task basis. Data about human behavior, such as collected in the studies presented within this thesis, helps to improve both production and interpretation of natural gestures. The following section provides a short review of gesture production in agents before approaches to the interpretation of gestures are considered.

3.1.4.1 Gesture Production

Indirect support of human-agent communication can be provided by creating realistic gaze patterns to support the agents' believability. Thus gaze plays an important role in the design of ECAs (Torres, Cassell & Prevost, 1997). Vertegaal, Slagter, van der Veer & Nijholt (2001) derive implications for gaze behavior of ECAs in communicative situations from eye tracking studies of human conversations. Others, such as Lee, Badler & Badler (2002), create computational models for gaze pattern production in virtual agents based on data on natural eye movements. Similar to this work, Raidt, Bailly & Elisei (2007) created Hidden Markov Models to control gaze behavior suitable for the cognitive state of the agent, based on multimodal data including gaze recorded and labeled in previous studies. Three of the authors, Picot, Bailly, Elisei & Raidt (2007) used computer vision implemented an online analysis of the visual scene faced by the agent, followed by the generation of a saliency map to locate potential fixation targets. A different approach was followed by Torres et al. (1997) who modeled gaze behavior as a function of discourse structure and turn-taking. Lee, Marsella, Traum, Gratch & Lance (2007) went even further by implementing a model which drives gaze behavior depending on conversational state and cognitive state, and on visual context.

A *direct* improvement of communication would require the intentional use of gaze in ECAs. Raidt, Elisei & Bailly (2005) demonstrated that deictic

expressions using gaze pointing in a virtual avatar could speed up selection task performance of users compared to configurations without avatar or with an avatar producing misleading gazes.

In a user study on the reception of gestures produced by an anthropomorphic agent, Nobe, Hayamizu, Hasegawa & Takahashi (2000) report a high amount of gaze on gestures. Their subjects used gaze to attend to 70.4% of the gestures produced by the agent. This is more than expected for general face to face human-human interaction. Yet this effect might have been caused by participants concentrating on the agent alone and the agent producing only relevant gestures. Gullberg & Holmqvist (2002) hypothesized that only the absence of social pressure allowed for a high concentration on gestures in the video with synthesized gestures used by Nobe et al. (2000). In their study on human-human face to face interaction, Gullberg & Holmqvist (1999) found that only 8.8% of their participants' fixations were aimed at the gestures. Instead, participants maintained eye or face contact, unless the interlocutors performed the gestures in their peripheral gesture space or explicitly emphasized the gestures by gazing at them. Yet there are two kinds of gestures the subjects in both studies dominantly attended to: auto-fixated facial expressions and the concrete pointing gestures which are the topic of this thesis.

3.1.4.2 Gesture Production in the Agent Max

In the group of Ipke Wachsmuth, the ECA Max (see Figure 3.1) has been developed since 1999. Max has an anthropomorphic appearance and is capable of moving his head, eyes and upper limbs in coordination with lip-sync speech production in a similar way to humans (Kopp, 2003). This surface realization of multimodal utterances is driven by a specification language for multimodal utterances called MURML (Multimodal Utterance Representation Markup Language) (Kranstedt, Kopp & Wachsmuth, 2002). Listing 3.1 shows a MURML specification of the deictic expression "Take this bar!" with a manual pointing gesture towards a certain Bar_1.

Kranstedt (2007) developed the MREC algorithm to select a (not necessarily minimal) set of discriminative features for a given referent object and its context objects. The algorithm thereby takes into account verbal expressions and manual pointing. In a second step, the surface realization for the deictic expression based on the selected features is specified using MURML and delivered for execution by Max.

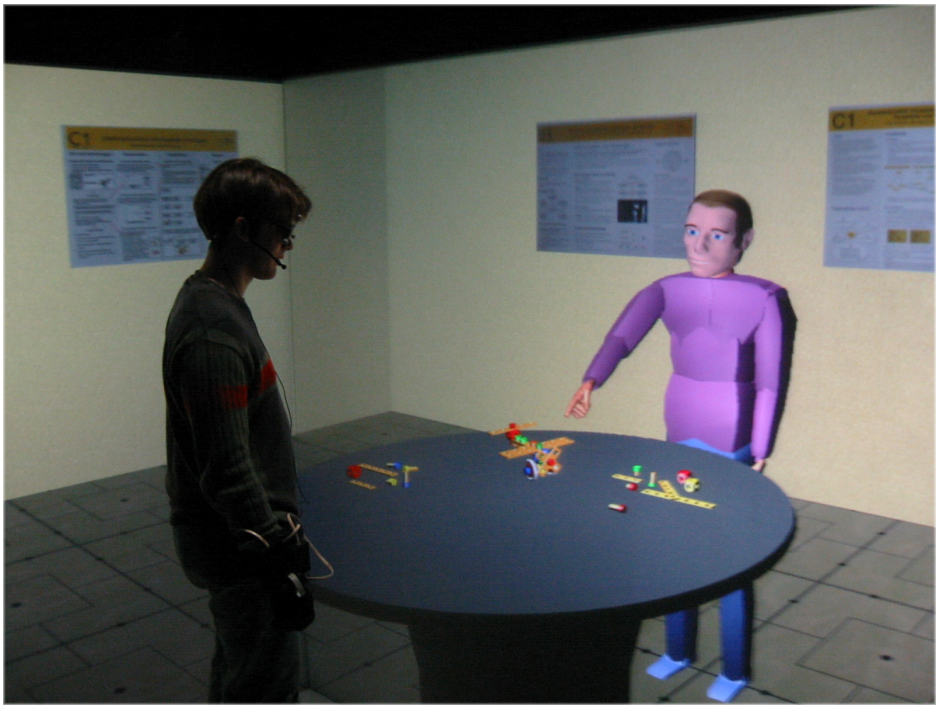


Figure 3.1: *The embodied conversational agent Max is able to produce and interpret multimodal utterances incorporating speech, gestures and gaze.*

During the evaluation of potential discriminative features, MREC needs to decide whether or not to use a manual pointing gesture. For this decision, the discriminative power of the manual pointing gesture has to be determined given the current context objects. The evaluation is based on a pointing cone model (see Figure 3.2) of the manual pointing gestures' scope. The development and parameterization of this cone model is based on data cooperatively gathered in a study on manual pointing (Kranstedt, Lücking, Pfeiffer, Rieser & Wachsmuth, 2006a), which is presented in Chapter 4.

Since then, the work on the production side of Max, gesture planning and realization, has extended to iconic gestures, starting from Kopp, Sowa & Wachsmuth (2004) and Kopp, Tepper & Cassell (2004).

3.1.4.3 Gesture Reception

In conversational interfaces, gaze input can be used as a "pointing device" in co-verbal expressions, but there are other applications as well. Vertegaal et al. (2001) showed in an experiment that interlocutors predominantly gaze

```
<definition>
  <utterance>
    <specification>
      Take <time id="t1"/> this bar <time id="t2"/>!
    </specification>
    <behaviorspec id="gesture_1">
      <gesture>
        <affiliate onset="t1" end="t2"/>
        <function name="refer_to_loc">
          <param name="refloc" value="$Loc-Bar_1"/>
        </function>
      </gesture>
    </behaviorspec>
  </utterance>
</definition>
```

Listing 3.1: *Specification of a deictic expression with speech and manual pointing gesture in MURML*

at partners they are speaking to (77% of the gaze to all participants while speaking) or listening to (88% of the gaze to all participants while listening). This makes gaze an excellent predictor to infer the conversational attention of an interlocutor. Oh, Fox, Kleek, Adler, Gajos, Morency & Darrell (2002) have taken up these findings and implemented their look-to-talk interface to trigger conversation with a conversational agent when being looked at. They report that this perceptual interface was preferably used over a manual push-to-talk interface in their study. Thus it seems that gaze pointing can also stand on its own, for example, expressing to the interlocutors "it is him, I am talking to". In a related application context, Nakano, Reinstein, Stocky & Cassell (2003) used the orientation of the user's head to approximate the direction of eye gaze. They exploited this information as a means to facilitate mutual understanding in a dialog with the ECA kiosk MACK.

Beyond the scope of deictic expressions, gestures are also interpreted for the purpose of turn-taking (Thórisson, 1997; Cassell, Bickmore, Billinghurst, Campbell, Chang, Vilhjálmsson & Yan, 1998; Traum & Rickel, 2002; Lessmann, Kranstedt & Wachsmuth, 2004), establishing joint attention (Pfeiffer-Lessmann & Wachsmuth, 2008) or intention reading, as the following examples document. Qvarfordt & Zhai (2005) demonstrated a tourist guiding system that adapts its output according to gaze patterns detected in the eye movements of the user. Morency, Christoudias & Darrell (2006) implemented a recognizer to interpret gaze patterns of users. This allowed the agent to infer whether the user was thinking about an utterance or waiting for the agent to

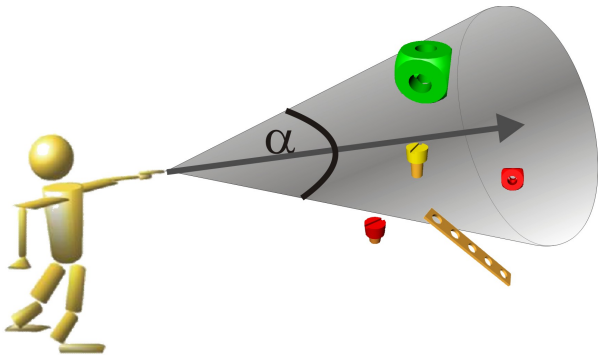


Figure 3.2: *One way to model the extension of pointing is using a pointing cone. The cone is a formalized way to account for the decreasing accuracy of pointing when pointing at distant objects.*

respond. Similarly, Eichner, Prendinger, André & Ishizuka (2007) used an online analysis of the user's gaze patterns to monitor for successful grounding, for example, after a deictic expression of an agent.

3.1.4.4 Gesture Reception in the Agent Max

Gesture reception in Max is handled via the frameworks ProSA and tATN (see 3.1.2 on page 37). This enables Max to recognize manual pointing and other types of gestures and interpret them together with speech input. Kopp et al. (2004) extended Max' capabilities to recognize and produce gestures to include shape-related iconic gestures. These are based on the Imagistic Description Trees described in Sowa (2006).

Max has only recently been enabled to interpret the user's eye gaze by Pfeiffer-Lessmann & Wachsmuth (2008), who used the DRIVE framework (see Chapter A) developed in this thesis *inter alia* to establish joint attention in a cooperative dialogue between Max and a human user.

3.2 Detecting Pointing in Gaze and Manual Gestures

As has been shown, gesture and gaze were introduced in HCI in the 1980s. Visionary ideas about multimodal interfaces that make use of gaze and gesture have been the driving force of the research since then. Progress has always

been tied to the development of appropriate sensing devices. In the following, a short review of the state of sensing devices and algorithms used to detect gaze and gesture pointing is given before investigating in detail the approaches taken to identify referent objects.

A well-known challenge of natural conversational human-computer interfaces is the *midas-touch problem*: a trigger is needed to activate referential use of the modality, otherwise every movement has immediate consequences and a controlled interaction is nearly impossible. An algorithm for detecting pointing gestures thus has to provide an answer to the question "When is the interlocutor pointing?"

In Section 2.3.1 it was argued that the typical function of a pointing gesture is that of a vector directed at the intended referent. Although a vector might not be sufficient to identify the referent, the origin and the direction of such a pointing vector are also relevant cues for all other models of pointing. An algorithm for detecting pointing gestures thus also has also to provide an answer to the question "From where and in what direction is the interlocutor pointing?"

3.2.1 Detecting Manual Pointing

Body movements, such as manual pointing gestures, are detected using motion tracking (sometimes known as motion capturing) technologies. A non-exhaustive list of tracking technologies includes inertial, magnetic, mechanical and optical tracking.

Inertial tracking systems use gyroscopes or accelerometers to detect motions, they are suited best for measuring velocity profiles and are not limited by occlusions. In fact, they are not dependent on external devices and are thus suited for motion tracking in the field. In order to determine absolute positions and orientations, the measurements need to be integrated over time, which is error prone due to the accumulation of noise and drifts. Examples are the systems offered by Xsens Technologies B.V. (2009).

Magnetic tracking systems consist of active markers that detect their position and orientation in a magnetic field, which is emitted by a base unit of the tracking system. A disadvantage is that they are sensitive to magnetic or conductive objects that will distort the magnetic field. Examples are the Flock of Birds from Ascension Technology Corpora-

tion (Scully & Blood, 1986) (see Figure 3.3 a) or the FASTRAK system from Polhemus (1994).

Mechanical tracking systems measure physical deformations. They do not require a straight line of sight and are suited to track hand postures without occlusions. They are also often found in combination with force-feedback systems. They are, however, restricted to smaller spaces. Examples of mechanical tracking systems for hand postures are the VPL Data Glove (Lanier & Zimmermann, 1986) or the CyberGlove (CyberGlove Systems, 1990) (see Figure 3.3 c).

Optical marker-based tracking systems need a line of sight between a sensory unit, typically operating in the infra-red domain, and either passive or active markers. Positions and orientations are derived by triangulation, so either two sensors need to detect one marker or one sensor needs to detect several unique markers to determine a position or orientation. Using passive markers on the body, these systems are relatively unobtrusive, but they have problems with occlusions. Examples are the OptiTrack system from NaturalPoint, Inc. (1997), the Impulse system from PhaseSpace (1994), the DTrack system from Advanced Realtime Tracking GmbH (2010) (see Figure 3.3 b) or the devices from Vicon Motion Systems (1984).

Optical computer vision-based tracking systems are the target of active research. They promise to be unobtrusive and flexible. Yet the systems available at the moment work best with a well-defined static background. Examples of research systems are the Stanford Markerless Motion Capturing System (Corazza, Mündermann, Chaudhari, Demattio, Cobelli & Andriacchi, 2006), commercial systems include Stage from Organic Motion, Inc. (2007).

For the studies in this thesis, body movements were tracked using an optical marker-based tracking system (DTrack). The DTrack system provides high accuracy, and tracks 3DOF (position) and 6DOF (position and orientation) targets. The 6DOF targets were used for an accurate tracking of the head, the elbows and the hands. In a pre-study, a CyberGlove was used to precisely track the more fine-grained movements of the fingers. But in the full study, a self-made glove using 3DOF targets from the optical tracking system was used; details will be provided in Chapter 4.

To answer the "when" question, a manual pointing gesture can be identified in the motion tracking data if at least two constraints hold:

1. The pointing hand has a typical shape, such as depicted in Figure 2.3

(a) Magnetic tracking, Flock of Birds from Ascension Technology Corporation

(b) Optical tracking, DTrack/Qualisys from Advanced Realtime Tracking GmbH

(c) Mechanical tracking, CyberGlove

Figure 3.3: *Manual pointing gestures are typically tracked using a combination of magnetic or optical tracking for the position and orientation of the hand, and a glove-based tracker for the hand posture.*

2. The trajectory of the hand follows a typical velocity profile along the three main gesture phases (preparation, stroke, retraction). The most relevant phase is the stroke.

One way to detect manual pointing gestures has been described by Latoschik & Wachsmuth (1998). They expressed the hand shape of a pointing hand using the declarative description printed in Listing 3.2. They expected the index finger to be in the process of elongation, already having passed a certain threshold and thus being nearly extended, while the other fingers are already curled beyond a certain threshold. The flection of the thumb is not considered. In the same paper, they suggested that this handshape, together with a pause in the acceleration of the forearm, are features that can be used to represent the stroke of a manual pointing gesture.

The "where" is typically identified with the tip of the pointing finger. The direction, however, is not as clearly defined. Intuitively, the pointing direction coincides with the extended pointing finger. We have coined this *Index-Finger Pointing (IFP)* (Kranstedt, Lücking, Pfeiffer, Rieser & Staudacher, 2006) to contrast it with an alternative approach for determining pointing direction. The latter, coined *Gaze-Finger Pointing (GFP)*, includes the direction of gaze, aiming over the tip of the index finger (see Figure 3.4). GFP is referred to as "occlusion selection" in HCI, whereas IFP is referred to as "raycasting selection". The two approaches can result in significant differences in the assumed pointing direction, as Figure 3.4 highlights.

```
Point_To := Elongating(Index) AND
            Elongate(Index, Min(threshI)) AND
            Elongate(Middle, Max(threshM)) AND
            Elongate(Ring, Max(threshR)) AND
            Elongate(Pinky, Max(threshP))
         => Action(Select Vector)

Where threshX is the threshold for the given attribute
and finger X.
```

Listing 3.2: *The specification of the handshape of manual pointing using the index finger, as provided by Latoschik & Wachsmuth (1998). Note that the flection of each finger is measured, and thus a value of zero represents an elongated finger.*

Kranstedt (2007) tested both approaches in a natural human-human interaction and was not able to clearly identify which one describes the pointing direction best. It could be that both approaches are used depending on other factors, for example, that gaze-finger pointing is used when pointing to more distant objects when the arm is raised high at eye level. Wingrave, Bowman & Ramakrishnan (2002) compared raycasting and occlusion selection in a fully immersive virtual reality task and found that users preferred occlusion selection as the faster and more accurate method in their study. Yet it is unclear how their findings extend to natural pointing in communication where the users perceive their own body.

3.2.2 Detecting Gaze Pointing

Eye movements are detected using so-called *eye trackers*. Today, most systems employ computer-vision techniques to detect pupil movements, but other techniques are available, for example based on muscle recordings. The vision-based systems predominantly operate in the infrared domain, where the pupil contrasts quite well with the surrounding matter. The most important feature of an eye tracker is its spatial resolution. While the foveal area extends about 2°, humans can differentiate between two points separated by about one arc minute (Velichkovsky et al., 1997). The application areas for eye tracking have different requirements on precision and temporal resolution. Accordingly, different kinds of eye-tracking systems are available .

Tabletop systems are fixed installations. The user has to put his head into an opening of the device. A chin rest is used to fixate the head during

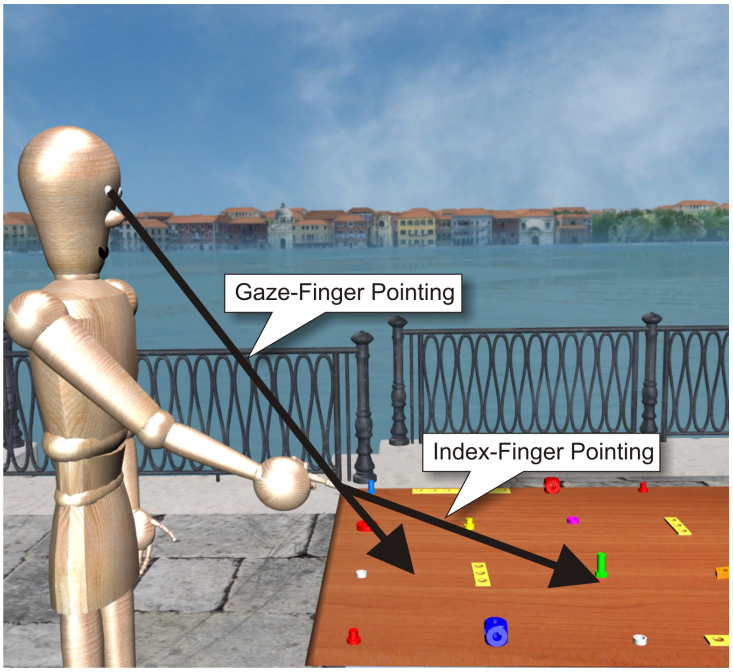


Figure 3.4: *The direction of a manual pointing gesture is not clearly defined. The graphic shows the intuitive interpretation along the direction of the extended finger, coined index-finger pointing. Gaze-finger pointing is an alternative suggestion that takes into account the line of gaze.*

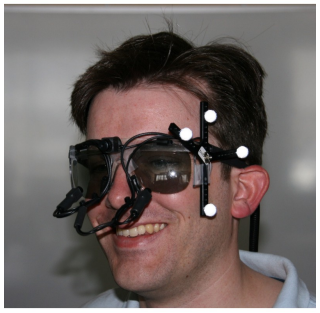
the study. These systems are very fast and precise. Their application domain is primarily psychophysiological research and they are less suited for HCI.

Remote systems are less obtrusive eye-tracking devices. The user is free to move his head, and the camera unit is placed on the table, for example below the display of a computer system used for stimulus presentation. The tradeoff for unobtrusiveness is a restricted area in which the head is allowed to move and a reduced precision both in time and space. Typical values are an arc-accuracy of below $0.5°$ and a sampling rate of $50-60\,\text{Hz}$, though further improvements are to be expected. Remote systems are predestined to be used in desktop-based HCI.

Head-mounted, stationary systems are split into at least two units. One, the camera unit (see Figure 3.5 a), is worn by the user, the other, the computer-vision system, is located in a black-box or a PC system. These systems offer a moderate to high precision in time and space. At the same time, the systems have to deal with artifacts induced by the

(a) SMI EyeLink I

(b) SMI iViewX



(c) Arrington Research ViewPoint PC-60

Figure 3.5: *A broad range of eye-tracking devices is available. The pictures show different head-mounted devices that may be used in Virtual Reality. The trend in development goes towards more lightweight devices, such as the SMI iViewX or the Arrington Research ViewPoint PC-60.*

movement, shifts of the head gear and perspective distortions. Most systems therefore include a local head tracking function operating in a spatial volume of 30^3 cm. Typical arc-accuracies are below 0.25° and typical sampling rates are 600 Hz. This system can be used in desktop-based HCI or in virtual reality installations where the user remains stable, probably seated.

Head-mounted, mobile systems are the more lightweight relatives of the stationary systems (see Figure 3.5 b and c). They are designed to be used in the field, for example for point-of-sale studies or in sports to analyze decision processes. Most units are equipped with a scene camera that video-records the area in front of the user. In an offline process, the fixations of the user are then overlaid over the recorded scene movie for further analysis. For online interaction, the eye-tracking system needs to be combined with a tracking solution to obtain the position and orientation of the user's head. Typical values are an arc-accuracy of 0.5° and a sampling rate of $50-60$ Hz.

For conversational interfaces supporting natural gestures, head-mounted mobile systems are the system of choice. They offer a moderate temporal resolution sufficient to detect relevant fixations and provide an accuracy that is better than that of human interlocutors. They also provide more freedom for movements, compared with remote systems, and are not affected by occultation during manual gestures.

Similar to manual pointing, identifying gaze pointing means

- to identify the time span during which the user is targeting his attention at something and
- to derive the direction of his gaze.

Considering the literature, two necessary preconditions can be identified. First, the eye should be fixating the target object, and second, the dwell time of the fixation should, according to Velichkovsky et al. (1997), be greater than 250 ms. However, these preconditions are not sufficient to separate voluntary gaze pointing from other gazes during sentence processing or involuntary gazes due to distractions. One way to deal with this is to require more explicit gaze pointing gestures, either by requiring an increased dwell time or by watching out for fixation patterns, such as a sequence of fixations starting on the interlocutor, then moving to the intended referent object and then moving back again on the interlocutor. Even then, false positives, i.e. detections of gaze pointing events that are in fact the result of different processes, are likely. In single-modality gaze-based interaction systems a typical threshold for fixation duration of about 500 ms is used to identify application-relevant gazes. Velichkovsky et al. (1997) recommend a threshold of $450 - 500$ ms. This general threshold can be reduced in highly-specific applications and with trained users, e.g. in gaze-typing systems. More robust approaches with relaxed requirements can be applied in multimodal interfaces, where gaze pointing is accompanied by speech and manual pointing. In such interfaces, the detected potential gaze pointing events are considered as hypotheses, and subsequent multimodal integration determines whether they are relevant or not.

Several algorithms have been developed to identify fixations in the raw eye movement data provided by the eye-tracking devices. Duchowski (2007) reviewed two algorithms proposed by Anliker (1976), the position-variance method and the velocity detection method. The position-variance method is based on the fact that during fixations the eye movement is relatively stable. The algorithm identifies a fixation if M out of N eye positions lie within an ε-area around the mean of the N positions. The values for M, N and ε have to be determined empirically. The minimum latency before a fixation can be detected is contingent on M. The velocity detection method is based on the fact that saccades have a higher velocity than the smaller eye movements during a fixation. The velocity during a small sample window of size N is measured and compared against a threshold V. If the velocity is smaller than V, the sample window is considered to belong to a fixation. Duchowski suggests combining both methods to bolster analysis and notes

that the velocity detection method usually has a lower latency. Based on the symmetric velocity profile of a saccade, one could also estimate the onset and the location of a fixation. This would require an eye-tracking device with a high temporal resolution.

Deriving the direction of gaze in 3D space is straightforward. It requires both the orientation of the eye relative to the user's head – this is provided by the eye tracker – and the position and orientation of the user's head in the world, which can be tracked using one of the techniques described above for tracking the hands during manual pointing. The HCI system needs to integrate the information from both tracking systems to determine the position and orientation of the eye in world coordinates, from which the 3D viewing direction can be derived.

3.2.3 Which one is faster, gaze or manual pointing?

If one has to decide which modality to use for object selection tasks, the speed of the interaction could be the decisive factor. Tanriverdi & Jacob (2000) compared object selection with gaze against object selection with the hand (the users were required to touch the objects) and found that gaze is significantly faster when selecting distant objects that would require additional movements to select manually. Cournia, Smith & Duchowski (2003) came to contradicting results when using a vector-based object selection algorithm for both modalities. They demonstrated significantly faster object selection using manual selection than when using gaze.

These seemingly contradictory findings might be the product of different ways of handling the "when" question. For gaze-based interaction, both studies required high dwell times before an object was selected. Manual selection, however, was triggered with the press of a button in the study of Cournia et al. (2003), while Tanriverdi & Jacob (2000) used a dwell time which was carefully adjusted to the dwell time used for the gaze-based selection. Thus, the results might have been different if Cournia et al. (2003) had either used a button to trigger gaze-based selection as well, or if they had used a dwell time for both of their selection techniques. In general, gaze can be expected to reach the target earlier than any manual interaction, but a certain dwell time needs to be taken into account to be sure that a target is fixated intentionally.

3.3 Interpreting Pointing

In Chapter 2 several informal, qualitative descriptions of pointing gestures and their interpretations have been reviewed. It seems, that scientific discussion so far lacks a rigid formal approach to model pointing. In the following, an attempt is made to come up with such a formal model, to lay grounds for a data-driven approach to model pointing in the remainder of this thesis.

Once a pointing gesture $p \in \mathscr{P}$ has been detected, it needs to be interpreted $I(p)$ to identify its referent in the referential domain $\mathscr{D}$:

$$\begin{aligned} I_{ideal} : \mathscr{P} &\rightarrow \mathscr{D} \\ p &\mapsto r \\ \text{with } \mathscr{P} &= \{(\vec{o},\vec{v}) | (\vec{o},\vec{v}) \in \mathscr{R}^3 \times \mathscr{R}^3\} \end{aligned} \tag{3.1}$$

The abstract representation of a pointing gesture used here is that of a tuple $(\vec{o},\vec{v})$ of the origin of the pointing gesture and its direction. The referential domain $\mathscr{D}$ is given by the situational context; it may contain all objects visible at the moment of pointing or else be further restricted by dialogue history, for example when the interlocutors have agreed on a certain type of objects as being relevant. In the following, this interpretation process is referred to as *dereferencing* pointing. The ideal interpretation I_{ideal} provides the one and only referent for each pointing gesture. In reality this is unlikely to happen. A more realistic version of I thus maps a pointing gesture to a *set* of possible referents, the *extension* of p.

$$\begin{aligned} I : \mathscr{P} &\rightarrow \mathscr{E} \\ p &\mapsto \{r | r \in \mathscr{E}\} \\ \text{with } \mathscr{E} &\subseteq \mathscr{D} \end{aligned} \tag{3.2}$$

It is also convenient to provide a ranking or, more generally, a weighting of the possible referents, sometimes further restricting the set of possible referents by a threshold. The dereferencing can then be decomposed into a selection S and a weighting W of possible referents.

$$I : \mathscr{P} \rightarrow \{x | x \in \mathscr{D} \times \mathscr{R}\} \tag{3.3}$$

$$S : \mathscr{P} \rightarrow \{r | r \in \mathscr{D}\} \tag{3.4}$$

$$W : \mathscr{P} \times \mathscr{D} \rightarrow \mathscr{R} \tag{3.5}$$

$$I : p \mapsto \{(r,w) | r \in S(p) \wedge w = W(p,r)\} \tag{3.6}$$

3.3.1 Dereferencing Pointing based on Direction

In the following section, several approaches for modelling the dereferencing function I will be presented. The approach most often found in HCI is the *vector extrapolation method*. It is a representative of the *direction-based approaches*, which have to deal with the problem that the distance of the referent pointed to is unknown and thus there can be, in principle, an infinite number of potential referents. To overcome this problem, heuristics or contextual information may be used to narrow down a set of potential referents. The second category consists of *location-based approaches*, which use either temporal or spatial integration to estimate the location of potential referents.

3.3.2 Vector Extrapolation

This method is technically often also referred to as *raycasting selection* and it is the standard picking operation of many interactive 3D graphics systems. As explained in Section 2.3.1, manual pointing is often associated with vector extrapolation from an origin $\vec{o}$, usually the tip of the pointing device, and an orientation vector $\vec{v}$. The function S can then be defined as an intersection between the geometries of the objects in $\mathscr{D}$ and the pointing vector given by $\vec{o}+d\vec{v}$, with d going from 0 to infinity. The mapping from the objects to their geometries is handled by G, all geometries are in $\mathscr{D}_G$.

$$\begin{aligned} S_{vector}(p) &:= \{r|(G(r)\cap\vec{o}+d\vec{v})\neq\emptyset\} \\ \text{with } G:\mathscr{D} &\rightarrow \mathscr{D}_G \\ \cap &:= \text{geometrical intersection} \\ r &\in \mathscr{D} \\ \vec{o},\vec{v} &\in \mathscr{R}^3 \\ d &\in \mathscr{R} \end{aligned} \tag{3.7}$$

The operation $\cap$ is a magic intersection function, which solves a system of equations describing geometries, and returns the solution if an intersection is found, or $\emptyset$ if the intersection is empty. The *weighting function* W for vector extrapolation which represents common uses is $W(p_x,r)=-d$, with d being the distance from the object to the pointing origin. According to this weighting function, objects closer to the origin are rated higher than more distant objects, and thus closer objects are preferred over distant objects.

Vector extrapolation for gaze pointing has been used by Duchowski, Medlin, Cournia, Murphy, Gramopadhye, Nair, Vorah & Melloy (2002) and

Barabas, Goldstein, Apfelbaum, Woods, Giorgi & Peli (2004). They use the position of the eye as origin and direct the pointing vector through the 2D fixation on a plane of projection which is provided by the eye tracking system. Both works identify the closest object as the only possible referent. Examples for uses of vector extrapolation for **manual pointing** are Lewis, Koved & Ling (1991) and their successors Codella, Jalili, Koved, Lewis, Ling, Lipscomb, Rabenhorst, Wang, Norton, Sweeney & Turk (1992), who used it in their "Rubber Rocks" game.

3.3.3 Shape-based Approaches

The vector-based approach can be generalized to *shape-based approaches* if G is extended to support shapes (or geometries) for the pointing gesture as well.

$$S_{shape}(p) \quad := \quad \{r|(G(r) \cap G(p)) \neq \emptyset\} \tag{3.8}$$

In their study on raycasting selection, Wingrave & Bowman (2005) implicitly use a *cone-based approach* when accepting 10 degrees of angular error during selection. This is a common method to compensate angular errors within the model, a claim that is also supported by Wingrave & Bowman (2005). The geometry of a cone can be described as

$$\begin{aligned} G_{cone} : 0 \quad &\geq \quad \vec{y} \cdot \vec{v} - |\vec{y}||\vec{v}| \cos\phi \\ \text{with } \vec{y} \quad &= \quad \vec{x} - \vec{o} \end{aligned} \tag{3.9}$$

Besides the origin $\vec{o}$ and the orientation $\vec{v}$, the cone is specified by its aperture 2ϕ. In Wingrave & Bowman (2005), the aperture was 20° or $\phi = 10^\circ$.

Olwal, Benko & Feiner (2003) use shape-based approaches in their augmented reality framework, where they call them "SenseShapes". They use different geometries, for example a pointing cone for the hand, to model the region of interest for individual sensors. Their approach was detailed in Kaiser, Olwal, McGee, Benko, Corradini, Li, Cohen & Feiner (2003a) where the authors also provide more details on their weighting function. The weighting function can be based on several features: accumulated duration within the area of interest, stability (1 - number of enterings/maximum number of enterings for any object), visibility (proportion of the projected area of interest taken up by the object) and the ranking for center-proximity. Only features that are appropriate for a specific modality are used, for example visibility is not considered for manual pointing. For center-proximity two different features are calculated: the distance from the closest point of the object to the center

of the SenseShape, as well as the average distance of all points to the center of the SenseShape. The description in Kaiser et al. (2003a) focuses on the framework and an evaluation study and does not provide more details on the parameters of the cones used for gaze or manual pointing. It is interesting to note that during multimodal integration different combinations of the rankings are used, depending, for example, on signals in the speech channel. This approach is similar to the procedure for multimodal integration used in this thesis, which is described in Section 6.5.

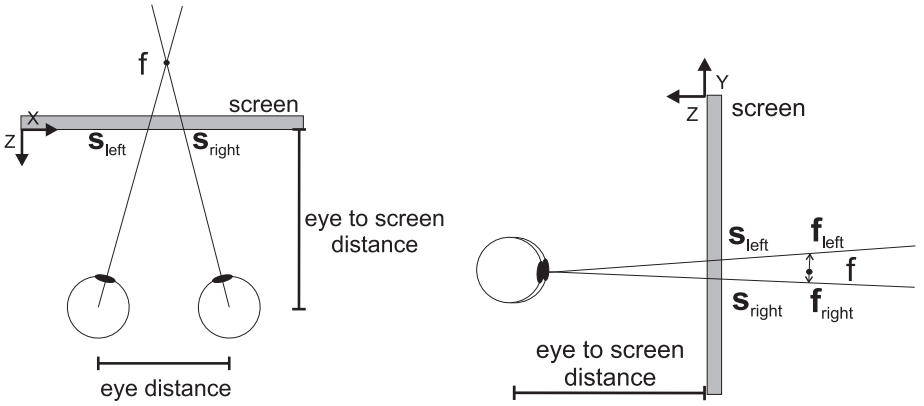
Shape-based Approaches for Manual Pointing An interesting shape-based approach has been reported by Barakonyi, Prendinger, Schmalstieg & Ishizuka (2007). They turn the problem of inaccurate pointing upside down by associating dynamic selection volumes (boxes or spheres) with each object. These selection volumes are then used for a classic vector-extrapolation based selection. The volumes are updated in size depending on the objects' distances to the viewpoint and are adjusted to avoid overlapping. This is, in effect, similar to a cone-based approach, but induces higher computational costs, since proxy geometries for the selection process have to be updated in real-time for all visible objects.

3.3.4 Dereferencing Pointing based on Location

Several approaches exist that try to infer the location pointed to either within a single modality by temporal integration or between modalities by spatial triangulation or using holistic approaches. Knowing the exact location pointed to is helpful for differentiating between foreground and background objects. This is difficult in direction-based approaches, as the pointing rays will hit the background most of the time. Additionally, it enables one to point behind objects, especially in the case where foreground objects are transparent.

Triangulation

A straightforward approach to locate the point of regard of a fixation is *spatial triangulation*. Examples of successful applications of spatial triangulation are Duchowski et al. (2002) and Kwon, Jeon, Ki, Shahab, Jo & Kim (2006). They estimate the depth of the fixation by intersecting the optical axes of the two eyes converging on the target (see Figure 3.6). However, the two visual



(a) top view (b) side view

Figure 3.6: *Using spatial triangulation, the depth of a fixation can be calculated. However, usually the two visual axes of the eyes will not intersect in 3D space. Seen from one perspective (a, top view) this might be the case, but not if another perspective is taken (b, side view).*

axes will usually not intersect due to noise and inaccuracies of the eye tracker or in the natural sight of the participant.

The following equations assume a coordinate system with an origin between the eyes of the observer. Given the positions of the two eyes $\vec{a_{left}}$ and $\vec{a_{right}}$, as well as the fixations of both eyes $\vec{s_{left}}$ and $\vec{s_{right}}$ on the plane of projection, we can derive the following parameterized line equations $\vec{g_{left}}$ and $\vec{g_{right}}$ for the visual axes as follows:

$$\vec{g_{left}} = \vec{a_{left}} + \mu(\vec{s_{left}} - \vec{a_{left}})$$
$$\vec{g_{right}} = \vec{a_{right}} + \eta(\vec{s_{right}} - \vec{a_{right}})$$

The points $\vec{f_{left}}$ and $\vec{f_{right}}$ on both visual axes in Figure 3.6 b are the points with the lowest distance to the other axis. The point of fixation $\vec{f}$ then is the mean of $\vec{f_{left}}$ and $\vec{f_{right}}$.

This approach, though, has some disadvantages. First, the physical parameters such as the height, the disparity and the geometry of the eyes vary between users and would have to be measured for each person. Also, one of the eyes typically dominates the other, that is, this eye's fixations are likely to be more precise and accurate than those of the other. More generally, users may have

different behavioral patterns in their vergence eye movements. Together with device-specific systematic errors and noise in the angles measured by the eye trackers this will lead to differences between the real and the approximated visual line. These parameters are not taken into account by this algorithm. An accurate calibration procedure could help to estimate some of the parameters. But to get reasonable data, calibration may have to be repeated several times, which would make it a tedious procedure. As the maintenance of an accurate tracking requires a recalibration every time the eye tracker slips, this would soon be tiring. Section 5.6 presents data on accuracy and precision from a study where spatial triangulation was used to estimate the point of regard in 3D.

Temporal Triangulation The difference between spatial and *temporal triangulation* is that spatial triangulation uses several pointing rays targeted at a referent from different positions (different modalities) but simultaneously, whereas temporal triangulation uses the pointing rays from a single origin but at different, consecutive points in time. In principle, the same equations as for spatial triangulation can be used. The drawback of temporal triangulation is its increased latency. The system has to wait for shifts in the position of the origin of the pointing ray that are sufficiently different from the previous position. However, if only monocular eye tracking is possible or the point of regards can be calculated post hoc, temporal triangulation can be an option. Mitsugami, Ukita & Kidode (2003) used temporal triangulation to estimate the depth of gaze fixations, but only in an offline post-processing of eye-tracking data.

Holistic Approximation via a Parameterized Self-Organizing Map

An alternative approach to triangulation has been proposed by Essig, Pomplun & Ritter (2006) for determining the 3D point of regard. They used a Parameterized Self-Organizing Map (PSOM), a smooth high-dimensional feature-map (Ritter, 1993) that adapts to the viewing behavior of the user and the visual context. The PSOM learns the mapping between the 2D coordinates of the fixations on a display and a fixated 3D point of regard.

The idea is to replace the fixed mapping provided by the linear algebra triangulation with a flexible mapping provided by a machine learning approach. This mapping translates the 2D coordinates provided for both eyes by the eye tracker to a 3D coordinate describing a singular binocular fixation in 3D space. This mapping will have to be learned and thus will require user

interaction. The 2D calibration procedure required for the 2D eye-tracking software will thus be followed by a 3D calibration procedure using a 3D grid of points. Another requirement therefore is that the learning procedure is as smooth and fast as possible, as relearning will be necessary every time the eye-tracking device slips.

The PSOM is derived from the SOM (Kohonen, 1990) but needs less training to learn a non-linear mapping. It consists of neurons $a \in A$ with a reference vector w_a defining a projection into the input space $X \subseteq \mathbb{R}^d$. The reference vector is defined as $w_a = (x_l, y_l, x_r, y_r, x_{div})$ with (x_l, y_l) and (x_r, y_r) being the fixations on the projection plane measured by the eye tracker. As the horizontal distance of the fixations has a significant contribution to the determination of the depth, it is added as an additional parameter $x_{div} = x_r - x_l$ to w_a.

To train the PSOM, all 27 points of a three-dimensional $3 \times 3 \times 3$ calibration grid are presented consecutively, and the corresponding w_a are measured. From this, one can derive a function $w(s)$ mapping the coordinates of the 3D grid onto the reference vectors. For this, $w(s)$ is parameterized as follows:

$$\begin{aligned} w(s) &= \sum_{a \in A} H(a,s) \cdot w_a \\ \text{with} \qquad H(a,s) &= 1 \ \text{ for } s = a \\ H(a,s) &= 0 \ \forall s \neq a; s, a \in A \end{aligned}$$

In this case, A is a grid of $3 \times 3 \times 3$ with 27 neurons

$$A = \{a_{xyz} | a_{xyz} = x\vec{e}_x + y\vec{e}_y + z\vec{e}_z; \qquad x, y, z \in \{0,1,2\} \quad \}$$

$$H : A \times \mathbb{R}^3 \to \mathbb{R}$$

To meet the required conditions, H is decomposed according to the product ansatz:

$$H(x\vec{e}_x + y\vec{e}_y + z\vec{e}_z, s_x\vec{e}_x + s_y\vec{e}_y + s_z\vec{e}_z) = H^{(1)}(x, s_x) \cdot H^{(1)}(y, s_y) \cdot H^{(1)}(z, s_z)$$

For the one dimensional function $H^{(1)} : \{0,1,2\} \times \mathbb{R} \to \mathbb{R}$ the following holds:

$$\begin{aligned} H^{(1)}(n,s) &= 1 \ \text{ for } s = n \\ H^{(1)}(n,s) &= 0 \ \forall s \neq n; s \in \mathbb{R}, n \in \{0,1,2\} \end{aligned}$$

As n can only take three different values, three cubic polynomials can be found, matching the requirements:

$$H^{(1)}(0,s) = \frac{1}{2}s^2 - \frac{3}{2}s + 1$$
$$H^{(1)}(1,s) = -s^2 + 2s$$
$$H^{(1)}(2,s) = \frac{1}{2}s^2 - \frac{1}{2}s$$

Thus $w(s)$ is constructed in such a way that the coordinates of the 3D grid can be mapped to the 2D positions of the fixations. To find the fixation one has then to find the solution of the inverse function numerically using gradient descent, which is done in the network's recurrent connections.

Essig et al. (2006) tested their approach in a desktop setting on a static anaglyphic stereo projection with dot-like targets. In their setting, the PSOM approach reduced the tracking error to 45% of the error produced by the geometric approach.

In the study presented in Chapter 5.6, a PSOM approach is compared to a geometric approach in a more realistic scenario with virtual objects in a desktop virtual reality scenario using shutter-glasses.

3.4 Integrating Multimodal Deixis

This thesis concentrates on the processes and models required between the detection of a gesture and the identification of its extension. The next step in processing multimodal deictic expressions is the integration of the contributions of the individual modalities. In the following, a selection of technical approaches to multimodal integration is briefly reviewed with the aim of deriving requirements which the results provided by the interpretation of the pointing gesture have to meet.

Koons, Sparrel & Thorisson (1993) created a 2D interface allowing for simultaneous speech, gaze and gesture input. The multimodal integration was achieved using frame-based representations created by modality-specific parsers. Their first prototype concentrated on deictic gestures (manual and gaze pointing) in an application scenario where objects had to be placed and moved on a map. In a second prototype (Sparrell & Koons, 1994), 3D objects were manipulated and iconic gestures were interpreted as well.

Latoschik (2002) used augmented transition networks to integrate speech and gesture. The dereferencing of individual modalities is done using SpaceMaps, that feed ranked lists of possible referents into the ATN. An extended version interpreted multimodal dereferencing as a fuzzy-based constraint satisfaction problem (Pfeiffer & Latoschik, 2004).

Kaiser, Olwal, McGee, Benko, Corradini, Li, Cohen & Feiner (2003b) demonstrate how mutual disambiguation greatly enhances the referencing process in the multimodal—speech and gesture—case, using cone-based object intersections for gestures. Their multimodal integration unifies typed feature structures which are constructed from the ranked lists of possible referents provided by the modality specific pointing modules. The unification is done in a generalized chart parser.

The aforementioned approaches to multimodal integration seem to be compatible with a basic set of information regarding the extension of an individual pointing gesture: a list of candidate objects, possibly ordered according to their relevancy. The approaches followed by Latoschik (2002) and Kaiser et al. (2003b) explicitly also require a history of such lists of candidate objects to support the integration of asynchronous contributions. These requirements are compatible with the definition of the interpretation function given in Section 3.3. The selection of candidates is handled by S and the ranking can be provided by the weighting function W. The temporal aspect is not explicitly represented in the interpretation function, but it is an inherent aspect of the processes.

3.5 Summary

Gaze and manual pointing gestures have been topic of HCI at least since Bolt's work in the 1980s. They are often guided, for example by auditory, force or visual feedback, and they may require a more tool-like pointing usage, both in motion and timing. Interaction using eye gaze has been envisioned as early as interaction using manual gestures, but the moderate technical progress in this area has slowed things down pre-millennially compared to interaction with manual gestures.

The vision of a conversational human-computer interface which is able to understand natural pointing gestures with hands or eye gaze is within reach. A detailed account of current work on life-sized embodied conversational agents (ECAs) has been given. The ECAs of today are already able to

produce a variety of human communicative gestures and facial expressions. A review of the features that have been identified as relevant to improve the communicative functions of the ECAs on the production side has been presented. This review provided viable insights on the relevance of eye gaze and manual gestures for human-human or human-agent communication.

Three major processes for understanding pointing have been identified: the *detection* of the pointing gesture, the *dereferencing* process in which possible referents are identified (often, but not mandatory unimodally) and *multimodal integration*. This thesis is primarily concerned with the process of identifying referents. The process of detecting the pointing gesture precedes this step and defines the kind of input that is provided for the identification process. The interpretation process similarly has to provide viable input to multimodal integration and the basic requirements for this have been identified in this chapter.

Regarding the timing of *gaze pointing*, two algorithms for identifying fixations in raw eye movement data have been presented. For *manual pointing* a declarative description of a pointing handshape has been found which formalizes the description given in Chapter 2. Both contributions add valuable information on the *when*, which can directly be used in implementing the software framework.

Concerning the *where*-question, two alternative models for the direction of *manual pointing* gestures have been introduced: *index-finger pointing* and *gaze-finger pointing*. The index-finger pointing model covers the common conception of the direction of pointing presented in Chapter 2. The gaze-finger pointing model is based on the direction defined by the gaze aiming over the tip of the pointing finger. It has been found that manual pointing gestures have a technical equivalence in HCI, namely *raycasting selection* or *occlusion selection*. Both techniques are used for *object selection*, which is the HCI equivalent of pointing to objects.

Regarding the *which*-question, the model of a *pointing cone* has been introduced. The pointing cone has already been successfully used to evaluate the discriminative function of a pointing gesture in gesture production. As an important step, a formalization of models for the extension of pointing has been developed to summarize and unify the findings so far. On the conceptual level, *direction-based* and *location-based* approaches are distinguished. The *vector extrapolation model* and *shape-based models*, such as the pointing cone, are examples of *direction-based* approaches. Examples for location-based approaches are *spatial or temporal triangulation* or *holistic approaches*, such as the presented machine-learning algorithm based on PSOMs. Especially

the location-based approaches promise a high spatial accuracy and should be superior to the direction-based approaches. However, they will only work for gaze pointing, as they require at least two valid directions of pointing from different origins.

Details on several tracking technologies have been provided, not only to give an account of current devices for HCI, but also in preparation of the studies which will be presented in the following chapters. Tracking technology for an accurate tracking of gestures in 3D is one of the supporting columns of the scientific methodology which has been developed to address the remaining open issues on the *where* and *which* of pointing.

Chapter 4

Manual Pointing

In this chapter the questions on the *where* and the *which* of manual pointing are approached. To this aim, a study has been designed and conducted to assess the morphology of the referential space of manual pointing. Particularly accuracy and precision are measured to describe the quantitative aspects of manual pointing gestures. Accuracy thereby refers to the degree of concordance between a conducted pointing gesture and the ideal pointing gesture under the vector extrapolation model. Precision refers to the similarity of different pointing acts. In the focus of the study are two interlocutors interacting over a domain of possible referents. The aim was to elicit natural deictic expressions and gather quantitative data on the position and orientation of the index finger during manual pointing acts in particular.

In pursuing these theoretical questions there was also a methodological challenge. Different modalities needed to be observed with a high precision and accuracy in real-time. Standard procedures, such as the recording and annotation of videos did not provide satisfying data. This meant that different recording technologies had to be synchronized during the study and a large amount of heterogeneous multimedia data had to be collected. In preparation for the analysis, these recordings then had to be integrated, checked for quality and annotated manually. Especially quality control and annotation proved to be difficult with standard procedures and software tools. As a consequence, the Interactive Augmented Data Explorer, a framework for studies on multimodal interactions was developed (see Section 4.6), which combines established scientific procedures with innovative techniques and integrates well with the workflow of the study. The presentation of this framework constitutes a second core theme of this chapter.

The following Section 4.1 provides background information on the development of the study. The empirical questions are detailed in the subsequent Section 4.2, before the design and set-up of the study is presented. The aforementioned methodological challenges are presented in Section 4.5 and the developed solution is presented in Section 4.8. The results are presented in Section 4.9.

4.1 Deixis in Construction Dialogues

The following study on pointing has been a conjoint effort together with Alfred Kranstedt, Andy Lücking, Hannes Rieser, and Ipke Wachsmuth in the contexts of the projects B3, Deixis in Construction Dialogues, and C3, Processing Instructions, of the CRC 360. Researchers from at least three different perspectives, namely linguistic theory building (Andy Lücking and Hannes Rieser), speech and gesture production (Alfred Kranstedt and Ipke Wachsmuth) and speech and gesture understanding (Thies Pfeiffer and Ipke Wachsmuth) met in this enterprise to create an annotated corpus of pointing games to lay grounds for their research. Andy Lücking's main contributions to this study were in the study design and the qualitative annotation of speech and gesture. Alfred Kranstedt focused on the study design, the conduction and the modeling of the scenario. The author's own contributions comprised the study design, the development of the technological framework, the analysis and the visualizations of the tracking data.

The author joined the group of the B3 project (Kranstedt, Lücking, Rieser, Wachsmuth) in July 2004, after half a year of cooperations with the former project members (Peter Kühnlein, Jens Stegmann). At this time, the project B3 had successfully conducted a study on co-verbal pointing that had been recorded using video cameras (see Figure 4.1). The analysis of this corpus had been achieved by annotating the video on a frame-by-frame basis to estimate the positions of the participants' pointing hands within each frame (see Lücking, Rieser & Stegmann (2004) and Kranstedt, Kühnlein & Wachsmuth (2004)). However, since pointing gestures are performed within 3D space, the 2D projection on the video film resulted in a loss of information that could not be compensated for. Thus, the results of this first study concerning the precision of pointing were not as reliable and precise as had been hoped.

With the technology developed in the course of this thesis, the afore-mentioned study was replicated using state-of-the-art tracking technology to precisely record the pointing movements of the participants. The motions of relevant body parts (fingers, hands or head) were captured in terms of absolute po-

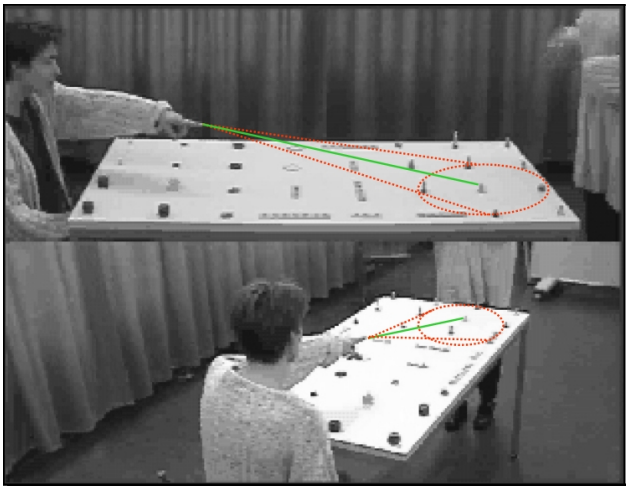


Figure 4.1: *A screenshot from a previous study on co-verbal pointing shows the difficulties in estimating the exact pointing direction (taken from Pfeiffer et al. (2006)).*

sitions and orientations in all three spatial dimensions without occlusions or perspective distortions. Moreover, this data was directly accessible for statistical analysis, without the need of manual annotations of finger positions. However, a manual annotation to identify the relevant phases of the pointing gestures was still required. Here again, virtual reality techniques were developed by the author that assisted in the annotation process by providing interactive visualizations of the recorded motion capture data and by making these recordings an object to manual annotation. The visualizations further helped to assess the quality of the recordings and to improve the overall quality of the gathered data.

In the end, an extendable methodological approach was developed that comprised audio, video and body movement recordings as well as human annotations. For this purpose, the IADE (see Section 4.6 or Pfeiffer et al. (2006)) was created, which supports both the recording of human-human or human-computer interactions, as well as the integration and visualization of synchronized multimodal recordings in a simulative setting. This is a novel experimental approach in the study of linguistic behavior.

4.2 Study Objectives

In the following, open questions posed in Chapter 2 and Chapter 3 are rephrased and substantiated, which form the starting points of the study presented in this section.

- *How does manual pointing to objects work?* There is a tradition of associating pointing with a vector and somehow deriving the referent object based on this vector. Yet, to the knowledge of the author, no substantial model for this process has been proposed. The study will collect precise data on manual pointing acts to approach this question.
- *How accurate is manual pointing?* Data on the recognition of pointing gestures has been presented in Chapter 2. But is it only a problem that can be attributed to the recipient if the interpretation of a pointing gesture fails? How accurate, in the first place, is the pointing gesture itself, considered from an objective perspective?
- *What defines the direction of pointing?* If the vector extrapolation model is applied, is the direction of the pointing gesture defined by the index-finger alone (Index-Finger Pointing) or by the direction of gaze of the producer aiming over the top of the index finger (Gaze-Finger Pointing)?
- *Is there an interaction between manual pointing and speech?* Are there differences in pointing gestures when they are produced co-verbally or not?

Throughout this thesis, the following definitions of accuracy and precision are used:

accuracy The term accuracy describes, how close each value obtained using a certain measurement system is to the real value. If the accuracy of a measurement system is high, then the measured values are close to the real values. The lower the accuracy of a measurement system is, the larger the errors get between the observed values and the real values. A low accuracy can be the result of a systematic error in the measurement system, e.g., a wrong assumption in the underlying model.

precision The term precision describes, how close the values obtained using a certain measurement system are when a specific real value is measured repeatedly. If the measurement system has a high precision, repeated measurements of the same real value provide measured values that are close together. However, they do not necessarily have to be close to

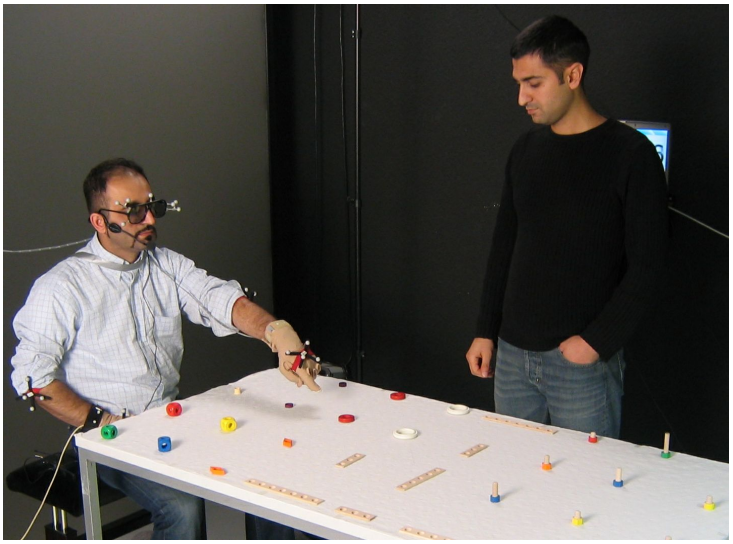


Figure 4.2: *Snapshot of a session in the study. The description giver, who is sitting on the left, is pointing at objects on the table. The object identifier, here on the left-hand side of the description giver, tries to identify the referent.*

the real value (which is described by the accuracy of the measurement system). If a measurement system has a low precision, repeated measurements will show large differences. If a system has a low precision, increasing the sample size could help to increase precision by averaging over several measurements.

If precision and accuracy of a measurement system are high, the system is said to be valid. Thus, when searching for models to describe manual pointing, one strives for such models that build the basis for a valid measurement system which offers high precision and accuracy. In this sense, the terms accuracy and precision are also used when referring to the quality of the models.

4.3 Study Design

The empirical study involved two participants for each trial who were engaged in a restricted object identification game. Each participant was assigned a certain role; one was the *Description Giver (DG)* and the other the *Object Identifier (OI)*. The game was designed in such a way as to elicit deixis using manual pointing gestures, either co-verbal or standalone. The game was repeated in two trials with differently positioned objects and in random order. In one trial the DG was allowed to use speech and manual gestures

(henceforth Speech and gesture trial (S+G trial)) and in the other trial the DG was restricted to use manual gestures only (Gesture-only trial (G trial)).

The interaction between DG and OI was restricted to avoid uncontrollable negotiation processes. Within each trial, 32 objects were demonstrated in a controlled order by the DG and identified by the OI along the following steps:

1. start of a new identification game
2. the object to demonstrate was presented to the DG on a display (M1 in Figure 4.4) via remote control
3. DG referred to the object on the table (S+G trial or G trial)
4. OI identified the possible referent using a pointing-stick
5. DG gave restricted feedback (*yes*/*no*)
6. in both cases, the identification game terminated and the participants began with step number 1 again, until all 32 demonstrations were completed

The procedure was explained to both interlocutors at the beginning of the session until all questions had been answered. The task was easily understood and there was no need for repetitions. While participants were asked in the G trials to generally use manual gestures to demonstrate the objects, they were not restricted to use a certain type of gesture and also the experimenter made no demonstrations, so as not to bias the genuineness of the participant's pointing behavior.

4.4 Domain of Possible Referents

Both OI and DG were located around a real table (70 cm × 155.5 cm) with 32 parts of a Lorentz Baufix toy airplane, the experimental domain. The objects' centers were lined up on an underlying grid, ensuring that they are laid out equidistantly (Figure 4.3).

With respect to the measurements of pointing accuracy that will be presented later, some data on the perceived layout of the objects on the table from the perspective of the DG are provided in this paragraph. The distance between the centers of the objects of the same row was 20 cm, the distances between the DG and the rows are shown in Table 4.1. The table also shows estimates of the angular differences the DG could perceive between the rows. The angle

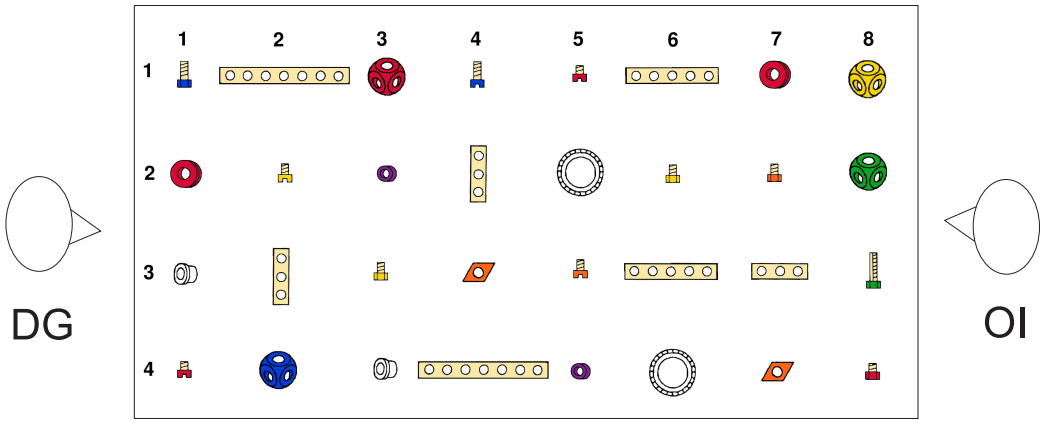


Figure 4.3: *The domain of possible referents was divided up into 8 rows and 4 columns, counted from the perspective of the description giver (DG) sitting to the left. It covered an area of* $70\,cm \times 155.5\,cm$*. In the study, the object identifier (OI) stood on the right side of the table. In the PDF version of this paper, a click on the image loads a 3D model of the table and the objects of the domain (Acrobat Reader might be required for that).*

$\gamma = \arctan \frac{h}{x}$ is specified between the horizontal line of sight of the DG and the direct line of sight straight to the specific row. These angles depend on the height of the description givers, so the table provides values for two exemplary heights, 50 cm and 70 cm, measured in seating position, with the surface of the table at 0 cm. Note the differences in angles between different rows ($\delta_{r,r+1}$ in Table 4.1): they specify the minimal angular distance between two objects as perceived by the DG. The data in Table 4.1 can be used to estimate the maximum deviation a pointing gesture can have from the ideal pointing direction before it could be mistaken as pointing to a different object. If the origin of the pointing vector is at gaze position and the direction lies exactly on the line of sight to the center of the demonstrated object, then the error should be less than $\alpha = \frac{1}{2}\delta_{r,r+1}$. For a larger error, the pointing gesture could be interpreted as referring to a different object in the next row.

Table 4.1: Configuration of the pointing domain. *The distances x are given from the position of the DG, also the angles between the horizontal line of sight and the direct line of sight straight to the row, for two sizes of description givers (in seating position, measured from the table surface). In addition, the difference in angle between the current row r and the following row r+1 is given as* $\delta_{r,r+1}$.

row r	distance x	h=50cm		h=70cm	
		$\gamma_{50\text{cm}}$	$\delta_{r,r+1}$	$\gamma_{70\text{cm}}$	$\delta_{r,r+1}$
1	7.75 *cm*	81.2	20.2	83.7	19.3
2	27.75 *cm*	61	14.7	68.4	12.7
3	47.75 *cm*	46.3	9.9	55.7	9.8
4	67.75 *cm*	36.4	6.7	45.9	7.3
5	87.75 *cm*	29.7	4.8	38.6	5.6
6	107.75 *cm*	24.9	3.5	33	4.3
7	127.75 *cm*	21.4	2.7	28.7	3.3
8	147.75 *cm*	18.7	–	25.4	–

4.5 Data Acquisition

DG and OI were placed in the area of the TRI-SPACE virtual environment in the AI Lab at Bielefeld University to utilize its marker-based optical tracking system. The motion tracking system (Advanced Realtime Tracking GmbH, 2010) consisted of nine cameras positioned around a cube of 2.6 m × 2.6 m × 2 m (see Figure 4.4), in which the study was set up. Two video cameras were positioned to provide one perspective from the side (see Figure 4.5 a) and one from above the OI. Only the DG was motion-tracked. He was sitting on a stool equipped with carefully positioned markers for the tracking system, measuring arm, index finger, hand and head movements. All DGs who entered the analysis were right-handed. Speech was captured by the DG's headset. The whole set-up with the prepared DG can be seen in Figure 4.5 (a), a screen shot from our video recordings. The special gloves used to track the stretched index finger are displayed in Figure 4.5 (b).

In a pre-study, the optical tracking system was used for tracking head and arms only. The hands were tracked using a CyberGlove (CyberGlove Systems, 1990) to obtain the full configuration of each hand. After five sessions, however, it turned out that some participants felt impeded by the cable-bound gloves, which resulted in a robot-like use of the pointing hand. This hampered the naturalness of their pointing behavior. Thus the CyberGloves were replaced by self-made gloves built from lightweight and flexible gloves normally used

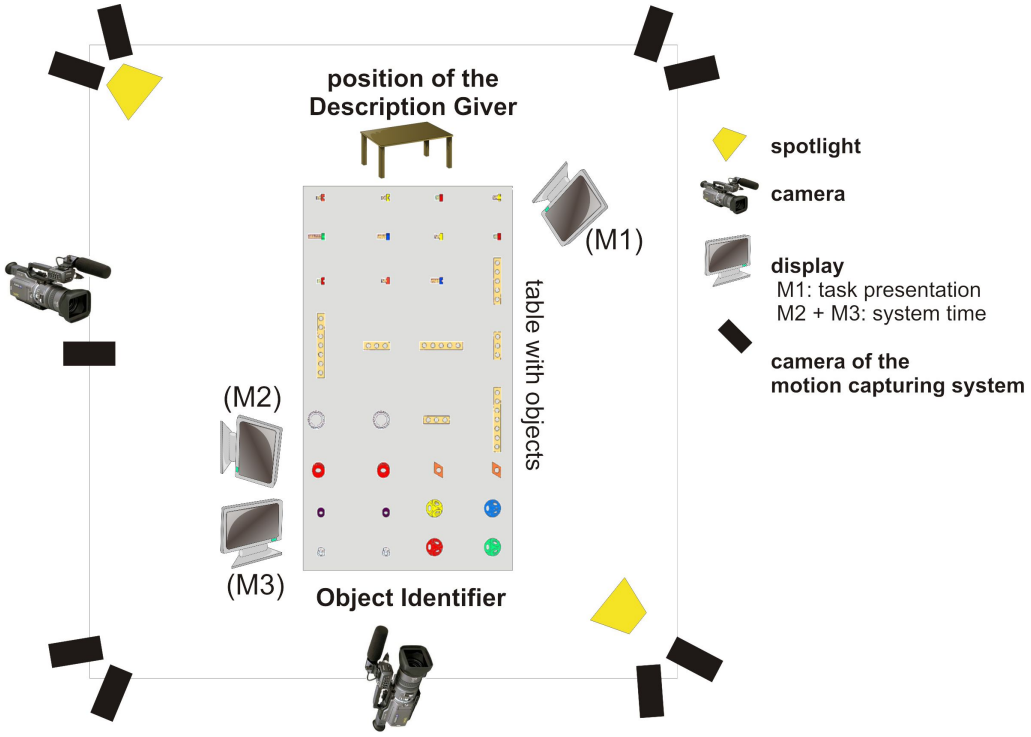
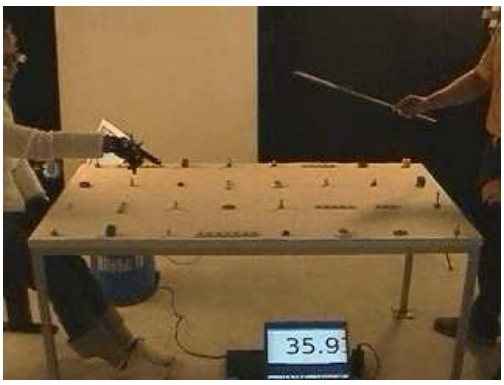


Figure 4.4: *The technical set-up of the study. The interaction area with the domain of possible referents on the table in the middle is surrounded by tracking cameras and video cameras to record every movement of the description giver.*

for golfing (see Figure 4.5 b), to which optical markers for the tracking system were attached. The markers were positioned on the second proximal and distal phalanges of each index finger. In the critical phase of manual pointing, the stroke, the index finger is partly extended, and thus the markers can be seen easily by the camera array. The markers on the phalanges provided a reliable pointing direction for Index-Finger Pointing even in cases when the finger was not fully extended. Subsequent tests showed that the new gloves eliminated the robot-like pointing problem.

Besides video and motion data, the utterances of the participants were recorded using microphones. This was relevant for the S+G trials as well as for the G trials, where the DG had to affirm a correct identification with a single “yes” or “no” for validation purposes. The audio recordings were mixed into the two video recordings on the fly to reduce synchronization overhead. Video and motion capture data were synchronized visually during a postprocessing step, using the time signal of the motion capturing system



(a) Video camera perspective

(b) Self-made soft gloves for optical tracking of the index finger

Figure 4.5: *(a) The study was recorded using two video cameras, here showing the side perspective, with the DG to the left. The system time needed for synchronizing motion tracking and video recording is displayed on a monitor on the floor. (b) Self-made gloves were used to track the DG's index finger.*

presented within each camera's perspective (see M2 and M3 in Figure 4.4 and the display in the lower part of Figure 4.5a).

4.6 The Interactive Augmented Data Explorer (IADE)

For the acquisition of the data during the trials and the integration and analysis of the multimodal data, the I*nteractive* A*ugmented* D*ata* E*xplorer* (IADE) was created (Pfeiffer et al., 2006). The data and process flow supported by IADE during the recordings is depicted in Figure 4.6.

During the recording of the study, IADE defined the primary time signal against which all other recording facilities were synchronized. While the motion capture system was directly connected to IADE, video recordings were synchronized by means of computer displays presenting the primary time signal to all camera perspectives (see M2 and M3 in Figure 4.4). Using the IADE framework, the motion capture data was integrated into a graph-based user model (see Figure 4.7). This model was then sampled at 25Hz and written to a file in the IADE Tracking Data file format. The video was recorded from the side and from above the setup using two camcorders with Mini-DV

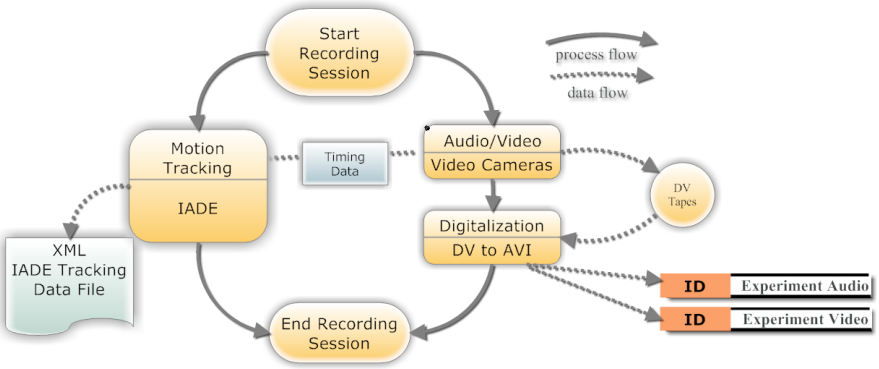


Figure 4.6: *The parallel recording of the multimodal data during the study was controlled by IADE.*

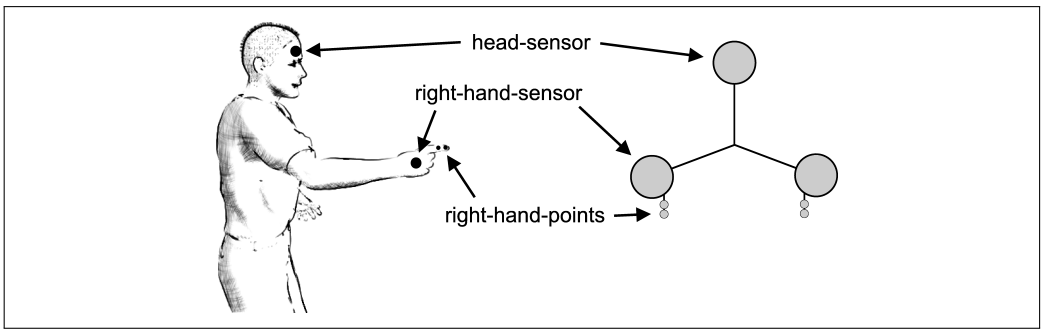


Figure 4.7: *A graph-based model of the description-giver was built by aggregating the data from the motion capture sensors. For each node on the right, the IADE system provided the exact position and orientation (except for the points) in space for each time step.*

cassettes. The audio recordings were made through a wireless head-set worn by the DG.

In a preprocessing step, the audio and video recordings were transferred from DV-tape to hard discs via firewire and transcoded and scaled from raw DV format to a size and format constrained by the requirements of the annotation software.

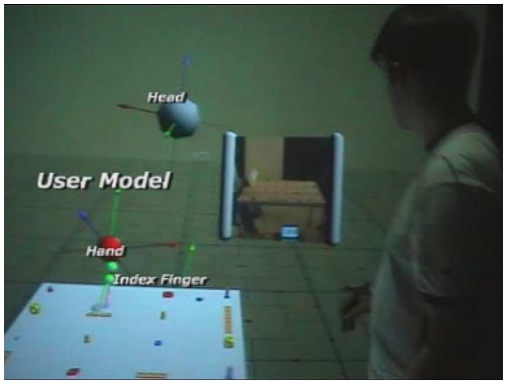
IADE was created as a fully immersive tool for the recording and simulation of multimodal data. It allows the user to literally enter the collected corpus data in virtual reality. During interactive exploration sessions, IADE displays the synchronized multimedia streams from the recordings as well as the annotations (see Figure 4.8 a). The user can enter the virtual space showing a

3D reconstruction of the study set-up and navigate freely through the setting, for example to switch between the perspectives of the DG or the OI at any time. In parallel, the audio and video recordings are available on movable and sizeable virtual panels that can be placed arbitrarily within the virtual world. In addition, the different annotation layers may also be visualized, so that the user is able to match the results from the motion tracking with the video recordings and the annotations. In this way, the quality of the recordings as well as of the annotations can be checked and corrected, if errors are found.

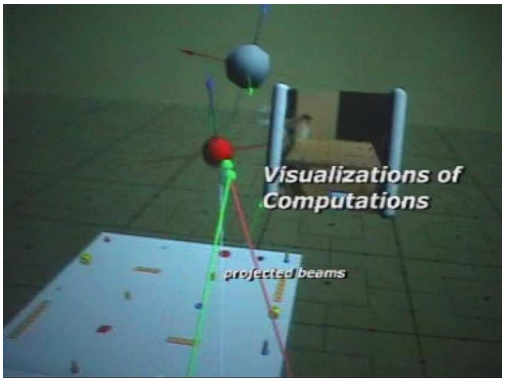
For the analysis of multimodal data and the iterative evaluations during data-driven modeling, IADE also features a powerful scripting interface. This interface can be used to create additional visualizations based on the data from the motion tracking and from the annotations. In the present study, this was used to visualize the pointing rays based on the tracked position of the index finger whenever a gesture's stroke was identified in the annotations (see Figure 4.8 b). In this way, the simulation can also be used to generate new data, such as the intersection points of the beams with the table's surface during the gesture's strokes. These can be easily calculated on the fly during the simulation.

For further analysis, IADE provides support to record videos of the interactive sessions. These can be made available for further offline analysis, for example for annotating different perspectives or visualizations of computations, thus closing the loop of iterative annotation, simulation and visualization.

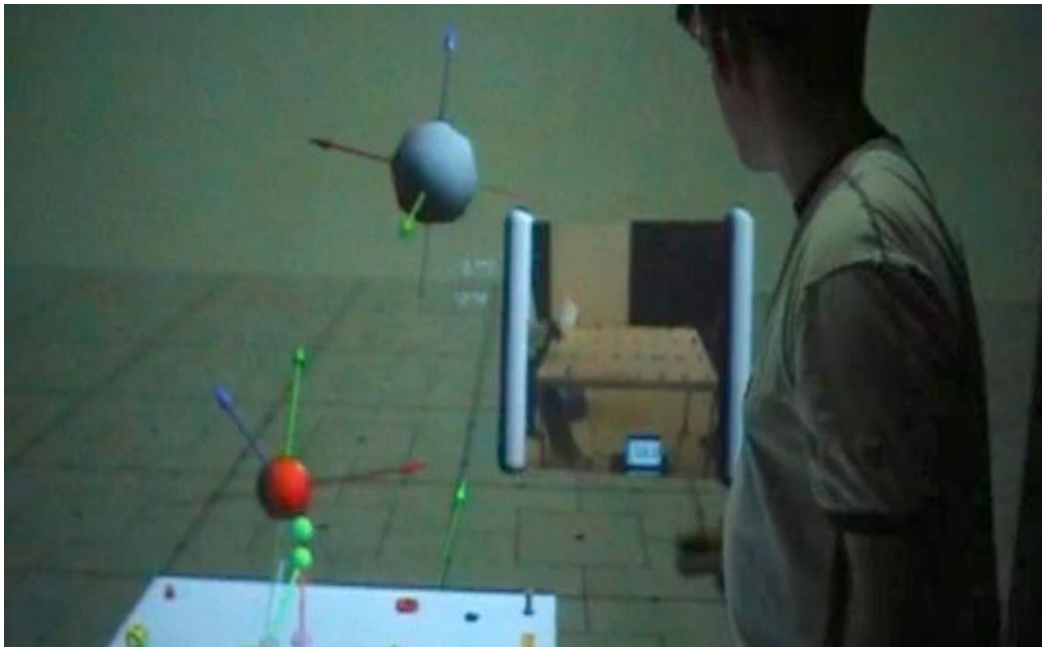
While IADE was primarily developed by the author of this thesis, over time, several people and projects contributed work: Alfred Kranstedt and Andy Lücking did the set-up modeling for the gesture study. Under the supervision of the author, Tobias Gövert assisted in the implementation of the data recording and the simulations and Nikita Mattar helped in implementing the manipulation techniques for the scalable video displays. IADE uses basic technologies developed in the Virtuelle Werkstatt (Biermann, Jung, Latoschik & Wachsmuth, 2002) and the PASION project (Pfeiffer & Latoschik, 2007) (floating video panels, see Figure 4.8).



(a) IADE visualizes relevant objects (*bottom left*), data from motion capturing (*left*) and video recordings (virtual panel, *right*).



(b) Added computations can be visualized in real-time. The example shows extrapolations of different types of pointing beams.



(c) IADE in action during the gesture study. In the PDF version, a click on the image starts the youtube video `http://www.youtube.com/v/21JD3uwWQLY`.

Figure 4.8: *IADE allows the researcher to interactively explore an integrated real-time simulation of all data gathered and annotated during the course of the study.*

4.7 Annotation

The video recordings and the data from motion capturing were reviewed, and the success, the demonstrated object and the identified object were annotated for each interaction game. In addition, the manual pointing gestures were identified and the critical interval of the stroke was marked. Only straight pointing gestures were considered; gestures simply following the morphology of the objects' geometries were ignored (for example drawing a circle or moving back and forth while pointing to a bar). Also, only one pointing gesture per game was considered, excluding exaggerated repetitions, for example when the DG was impatient. Three raters annotated 64 pairs of videos with 2048 demonstrations. Inter-rater agreement was assessed regarding semantic classification and the identification of valid gestures on selected videos, and a high level of agreement was attested (Kranstedt et al., 2006).

Besides IADE, Anvil (Kipp, 2001) was used for the manual annotation, supported by Praat (Boersma, 2001) for the transcription of spoken language in the S+G trials. The annotation of the manual gestures was restricted to the DG's first pointing act in each game. The main annotation layers were the following:

gesture.phase [*preparation*, *stroke*, *retraction*] the phases of the pointing gestures were identified according to McNeill (1992). Relevant for the later analysis is the stroke phase.

gesture.handedness [*left*, *right*] although only right-handed DGs took part, they still used the left hand in some cases, and this was annotated accordingly.

speech.transcription the DG's exact words were transcribed.

speech.number the number of words used in the S+G trials for one move was counted.

speech.quality [*shape*, *color*, *function*, *position*, *proxy*] an internal categorisation of aspects of the speech that were relevant for later analysis. The category *proxy* labeled taxonomically unspecified nouns, NPs or determiners, like "Ding" (*thing*) or "Das" (*that*) or "Dieses Teil" (*this thing*).

move.referent [γ] the internal identifier γ of the referent object prompted to the DG.

move.success [*yes*, γ] marked whether a move had been successful (*yes*); if the OI identified a false referent, its internal identifier γ was annotated.

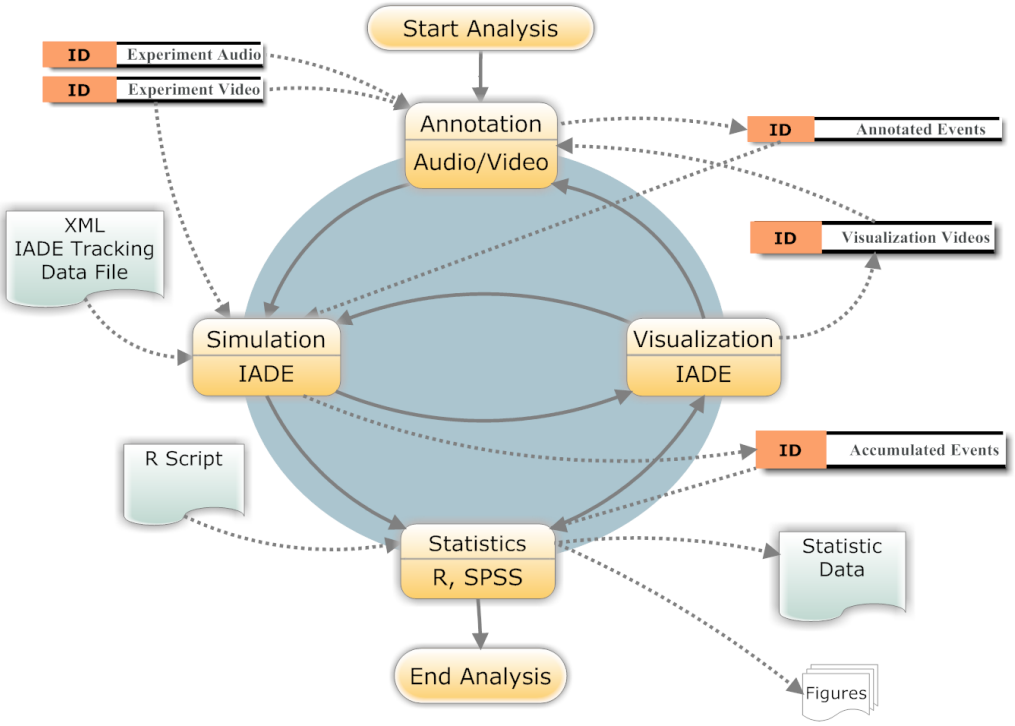


Figure 4.9: *Processing of recorded and transformed data was done iteratively: data was* **annotated**, *integrated and enriched in* **simulation** *runs, and analyzed by* **statistical** *processes. In parallel, an interactive* **visualization** *of the results allowed for qualitative analysis by human raters, for example doing cross-modality checks, which sometimes led both to re-annotations or to a refinement of annotation schemes.*

4.8 Simulative Analysis and Visualization with IADE

During the progress of the annotation and analysis, the data was iteratively integrated and evaluated using IADE. This process is depicted in Figure 4.9. After starting the analysis, the manual *annotations* were fed into IADE's *simulation* core, together with the *primary corpus data (XML IADE Tracking File)* consisting of the tracking data and the recorded videos. IADE features a scriptable compute kernel which can be used to interactively design data evaluations or extrapolations and to compute *statistics*. The statistics were then exported to R or SPSS.

The results of the simulation were then *visualized* in a virtual reality environment where they were explored interactively (see Figure 4.8) to control data quality and to visually verify the hypothesis generated by the different pointing models. The results of this analysis are presented in the following sections.

4.9 Results

Overall, data from 32 description givers and 32 object identifiers was recorded. The data was cleaned by removing cases where there had been a technical problem with the recordings during the trial and by removing cases with no pointing gestures. In one removed case, for example, the DG used a schematic approach by manually indicating column number and row number. After cleaning the data, a high-quality dataset consisting of 22 S+G trials and 22 G trials was selected from 25 description givers (19 females, 6 males) and 25 object identifiers (11 females, 14 males). For three G trials, pairings of interlocutors different from the S+G trials had to be chosen. The mean age of the participants, most of them students at Bielefeld University, was 26.02 (SD = 7.07). In the S+G trials, 514 successful demonstrations with manual pointing gestures were identified, compared to 443 in the G trials.

The quality of annotations and motion capturing was assessed using IADE by displaying the two perspective videos, the annotation data and the data from motion capturing simultaneously. Using this method, 957 high quality data records of demonstrations were identified. For most pointing gestures, several samples of tracking data during each stroke were collected. This data was reduced to the median per stroke, resulting in a final set of 957 postures, one for each annotated demonstration.

The organization of the domain of possible referents into a grid layout of 8 rows and 4 columns induced an analysis of the collected data that was also oriented on this design. This layout had been introduced for studies on co-verbal pointing in the DEICON project of the CRC 360 (Kühnlein & Stegmann, 2003) and the follow-up study presented here adhered to this scheme (see the related publications, authors in these publications are always in alphabetical order, (Kranstedt, Lücking, Pfeiffer, Rieser & Wachsmuth, 2006b; Kranstedt et al., 2006a; Kranstedt, 2007)). The following presentation of the results will do likewise and adhere to the grid layout model. Chapter 6 then presents a different topological model for the interpretation of the results, which is better suited for geometric analysis.

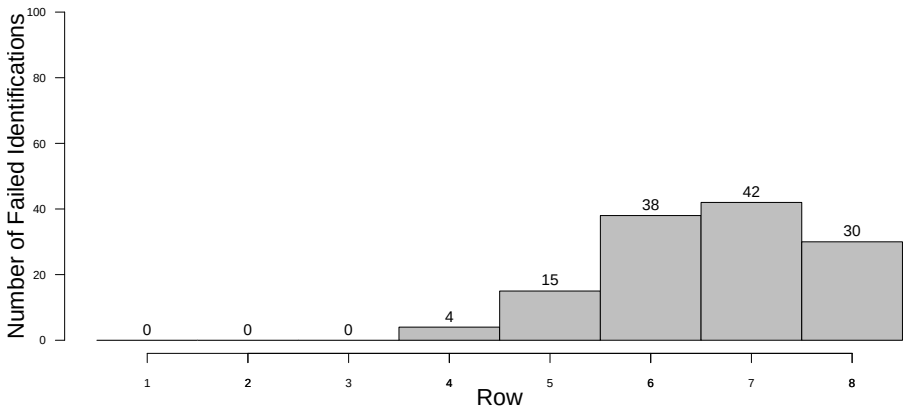


Figure 4.10: *The barplot shows the number of failed identifications per row for the G trials (there were no failures in the S+G trials). Failed identifications increase from row 4 on.*

4.9.1 Success of Manual Pointing

In the S+G trials, the OIs were able to identify the referents demonstrated by the DG perfectly, with a success rate of 99.8%. This was expected, as the objects in the domain have features (color, shape, type) that are easily distinguishable using speech. A different picture, however, was found in the G trials. The failed moves, i.e. moves where the OI did not manage to identify the demonstrated object, are shown per row in Figure 4.10.

The number of failures starts to rise beginning with row 4, which is 67.75 cm away from the description giver, and increases until row 8. The initial increase in failures around rows 4 and 5 can be interpreted as marking the border between proximal and distal pointing in our setting. The proximal area resides within easy grasping space of the seated description giver, while the rows beyond 5 do not. The drop in error rate in row 8 is surprising. After reviewing the videos it was noticed that some description givers exhibited a different pointing behavior when pointing to this border of the pointing domain: they exaggerated their gestures vertically to clearly indicate a reference to the last row, and differentiated only horizontally between single objects. This helped to reduce interpretation errors, as these objects then only had to be separated from the neighbors in their own row.

The distinction into proximal and distal areas for pointing gestures fits nicely into the dichotomy common in many languages for deictic expressions (*here* vs.

there). The distinctive feature is whether the self is included (near/proximal) in the area, or not (far/distal) (see Sennholz, 1985). For manual pointing, the grasping/touching area is an intriguing candidate for the proximal area, and everything beyond may be attributed to the distal area. This and the results shown in Figure 4.10 motivates splitting the pointing domain into a proximal area comprising rows 1 to 4, and a distal area, rows 5 to 8.

4.9.2 Applying the Vector Extrapolation Model

To investigate closer why the OIs had so much difficulty in identifying referents in the distal area, which was actually closer to where they were standing, one can ask how precise the pointing gestures from the DGs were. One way of answering this question is by applying the vector extrapolation model to the motion tracking data. It basically identifies the direction of the pointing gesture and can be used to test where the direction intersects with the pointing domain. This was done by simulating the recorded data with IADE, and the results are shown in Figure 4.11 for the S+G trials. For both variants of the vector extrapolation model, index-finger pointing and gaze-finger pointing, the resulting intersections of all 514 demonstrations are shown, for all objects and for all participants. The positions of the objects are marked as black squares. All pointing gestures targeted at a specific referent are drawn in the same color. While these dot-clouds are quite compact and stand out quite nicely in the proximal area, the dots alone would not have provided a good understanding of the distal area. Therefore, each cloud of intersections per object has also been approximated by an ellipse around 0.75% of the intersections.

Accuracy and Precision Very early in the analysis of the motion capturing data in IADE it was confirmed that pointing is, as expected, quite imprecise. For selected objects Figure 4.12 shows a different kind of visualization, called bagplots (Rousseeuw, Ruts & Tukey, 1999), of the intersections between the extrapolated pointing vector, here directed using the Gaze-Finger Pointing model, and the surface of the table during a pointing stroke. Both accuracy and precision decrease badly with increasing distance from the description giver, considering that row 8 is only 150 cm away.

To answer the question about accuracy and precision of manual pointing, an error measurement is required that captures the relevant discrepancy. Two such measurements have been used: *orthogonal distance* and *angular distance*. Orthogonal distance is defined as the distance between the center of the referent object and the pointing ray. Angular distance is defined as the angle between the pointing ray and an ideal ray, starting in the same origin but directed exactly to the referent object. The angle is measured in the origin of the pointing ray.

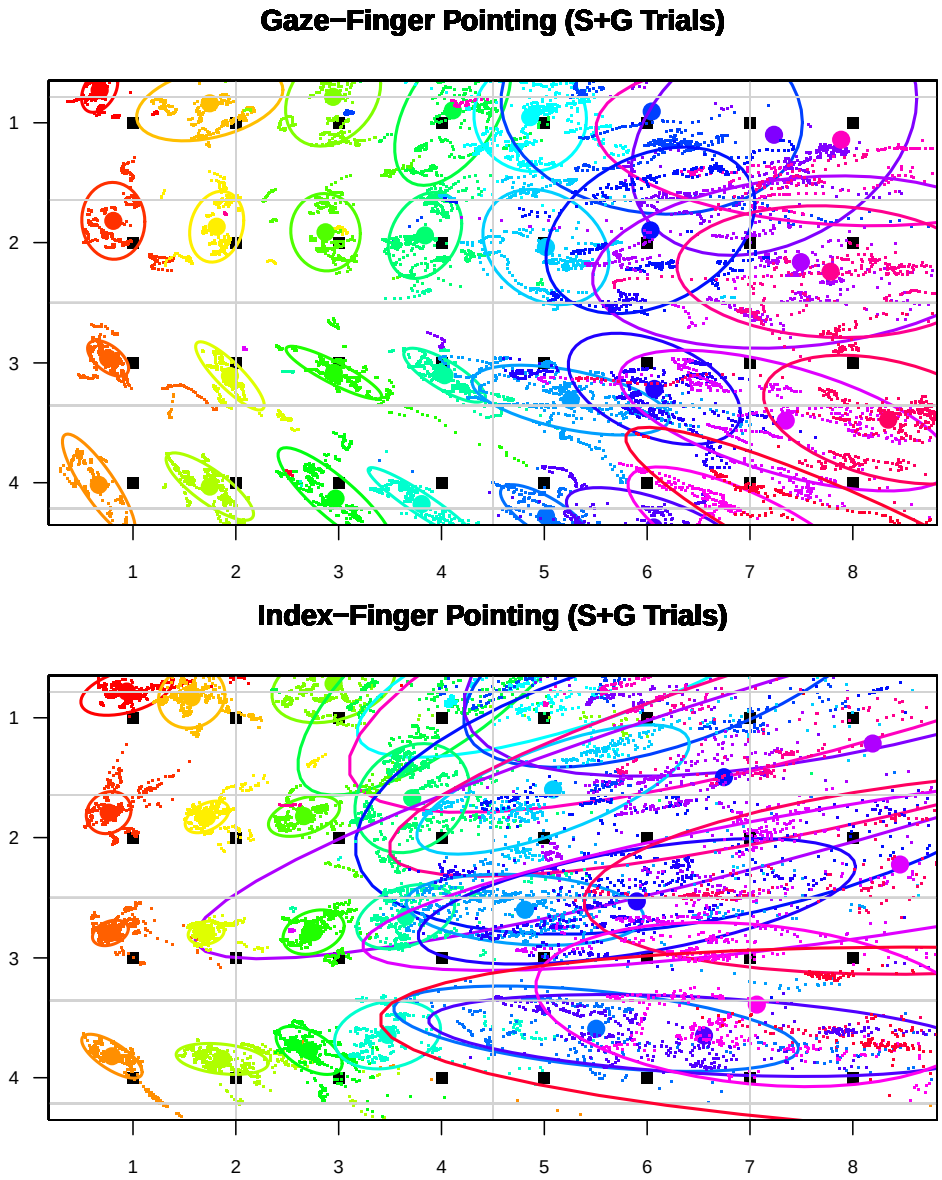


Figure 4.11: *If the vector extrapolation model is applied, the intersections of the pointing vectors with the surface of the table can be calculated. The graphics show these intersections for the S+G trials with different pointing directions, GFP (top) and IFP (bottom).*

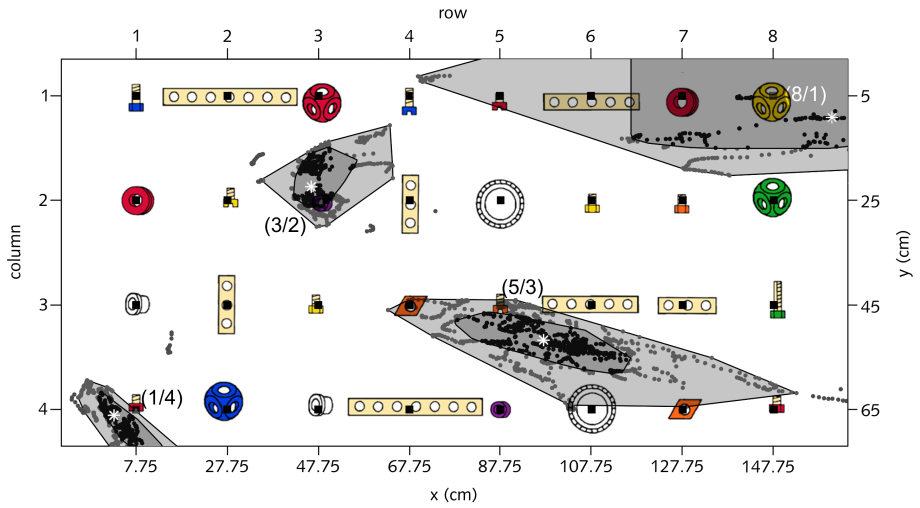


Figure 4.12: *For a selection of representative objects, the distributions of the intersections between the extrapolated pointing vector and the surface of the table are shown as bagplots for the S+G trials (50% of the points lay in the dark grey, 75% in the light grey area).*

The orthogonal measurement is more objective, as it does not take into account the perception of the description giver. Thus, the orthogonal distance will expectably be higher for objects more distant to the DG. This might be a better description of the difficulties the OI had in identifying referents. The angular distance abstracts away from the distance of the object to the DG. Given two distant objects, the angular distance approximates the visual distance of the objects, as perceived by the DG. In contrast to the orthogonal distance, the angular distance measured between two points decreases if the distance from the DG to the objects is increased. The angular distance better describes the accuracy as perceived by the DG.

> The angular error describes the accuracy of manual pointing perceived by the DG. The orthogonal error describes the objective error of manual pointing, which can also be perceived by the OI.

Using IADE, the distance errors were calculated over all demonstrations. The results are depicted in Table 4.2 for the Index-Finger Pointing and the Gaze-Finger Pointing model regarding the two distance measures. Given the spacing of 20 cm between the objects, a mean orthogonal distance above 10 cm already means that the pointing gesture could have been targeted at a neighboring object. The mean errors shown in Table 4.2 underline what has already been seen in Figure 4.11: manual pointing is extremely imprecise.

Table 4.2: Pointing Accuracy and Precision *Means and standard deviations for the orthogonal and angular measures. The table shows the data for both trials and the IFP and the GFP model. Angular errors are given in degrees, orthogonal errors in cm.*

r	S+G Trials errors per row							
	Gaze-Finger Pointing				Index-Finger Pointing			
	angular		orthogonal		angular		orthogonal	
1	36.3	±18.0	7.4	±4.2	30.1	±13.3	6.4	±3.5
2	23.5	±14.6	6.7	±5.0	24.4	±12.9	6.8	±3.6
3	16.4	±11.7	6.2	±4.7	19.3	±10.4	7.1	±3.8
4	13.3	±13.4	8.2	±10.5	19.0	±12.0	11.3	±10.1
5	7.3	±3.8	6.9	±4.0	13.2	±7.0	12.2	±6.1
6	9.0	±7.2	11.8	±10.3	12.0	±5.3	15.6	±8.0
7	8.8	±7.7	14.7	±14.1	11.8	±6.2	19.3	±10.9
8	9.2	±9.0	19.1	±19.2	10.8	±4.9	22.2	±11.0
r	G Trials errors per row							
	Gaze-Finger Pointing				Index-Finger Pointing			
	angular		orthogonal		angular		orthogonal	
1	32.1	±15.7	6.0	±4.7	34.1	±11.8	6.1	±2.7
2	23.6	±13.8	5.1	±3.6	32.0	±10.2	6.8	±2.5
3	19.6	±18.2	6.6	±10.8	26.4	±11.0	8.1	±8.6
4	17.5	±15.3	7.8	±9.1	21.6	±11.7	8.2	±5.3
5	13.0	±9.2	7.3	±6.2	14.8	±7.2	7.5	±3.3
6	12.3	±9.1	9.8	±8.8	15.0	±5.2	11.7	±6.1
7	11.3	±6.0	13.4	±7.5	11.0	±4.9	12.8	±5.2
8	11.1	±5.7	18.9	±9.3	11.7	±6.3	19.6	±9.1

It can also be seen in the data for row 8 that orthogonal errors are still increasing, while Figure 4.10 shows the drop in identification failures for the same row. This underlines the explanation given above that this effect is not due to a sudden boost in pointing accuracy, but to a borderline effect: referent objects in row 8 only have to be distinguished from other objects in row 7 and within row 8. There is, however, no row 9, and this fact was used by the DGs to exaggerate their gestures, thus reducing the chance that the OI misinterpreted the pointing gesture as targeting an object from row 7.

Index-Finger Pointing or Gaze-Finger Pointing One initial question driving the study was whether the direction of the pointing gesture is approximated better with the GFP or the IFP model. In addition to Table 4.2,

Figure 4.13 depicts the medians of the results for both models side-by-side. A statistical analysis (paired t-test, $\alpha = 0.05$) reveals that for the S+G trials IFP produces a significant larger angular error in rows 3 to 7 than GFP. The errors are significantly lower for the first row, and no significant difference can be found for rows 2 and 8. Considering the orthogonal errors, rows 4 to 6 show the same pattern of significantly larger angular error for IFP than GFP; while in row 1 IFP produces significantly lower errors than GFP, the differences in all other rows are not significant.

The results for the G trials are less clear. Here IFP produces a significant larger angular error for the rows 2, 3 and 4, and a significant larger orthogonal error for rows 2 and 3. All other differences are not significant. Summarizing these findings, in the S+G trials, GFP produces significantly lower angular errors than IFP for rows 3 to 7 (most of the distal area). In the G trials, GFP is only superior over IFP in rows 2 to 4. Overall, GFP approximates the ideal pointing direction better than IFP when using angular error measurements.

Pointing Accuracy in S+G trials vs. G trials When comparing the results between the trials, the angular errors for IFP are significantly lower in rows 2, 3 and 6 in the S+G trials than in the G trials. However, the orthogonal errors of IFP are significantly greater in rows 4 to 7 in the S+G trials. There are less significant differences for GFP. In the S+G trials, the orthogonal error of GFP is significantly greater in row 2 than in the G trials and in row 5 the angular error is significantly lower in the S+G trials than in the G trials.

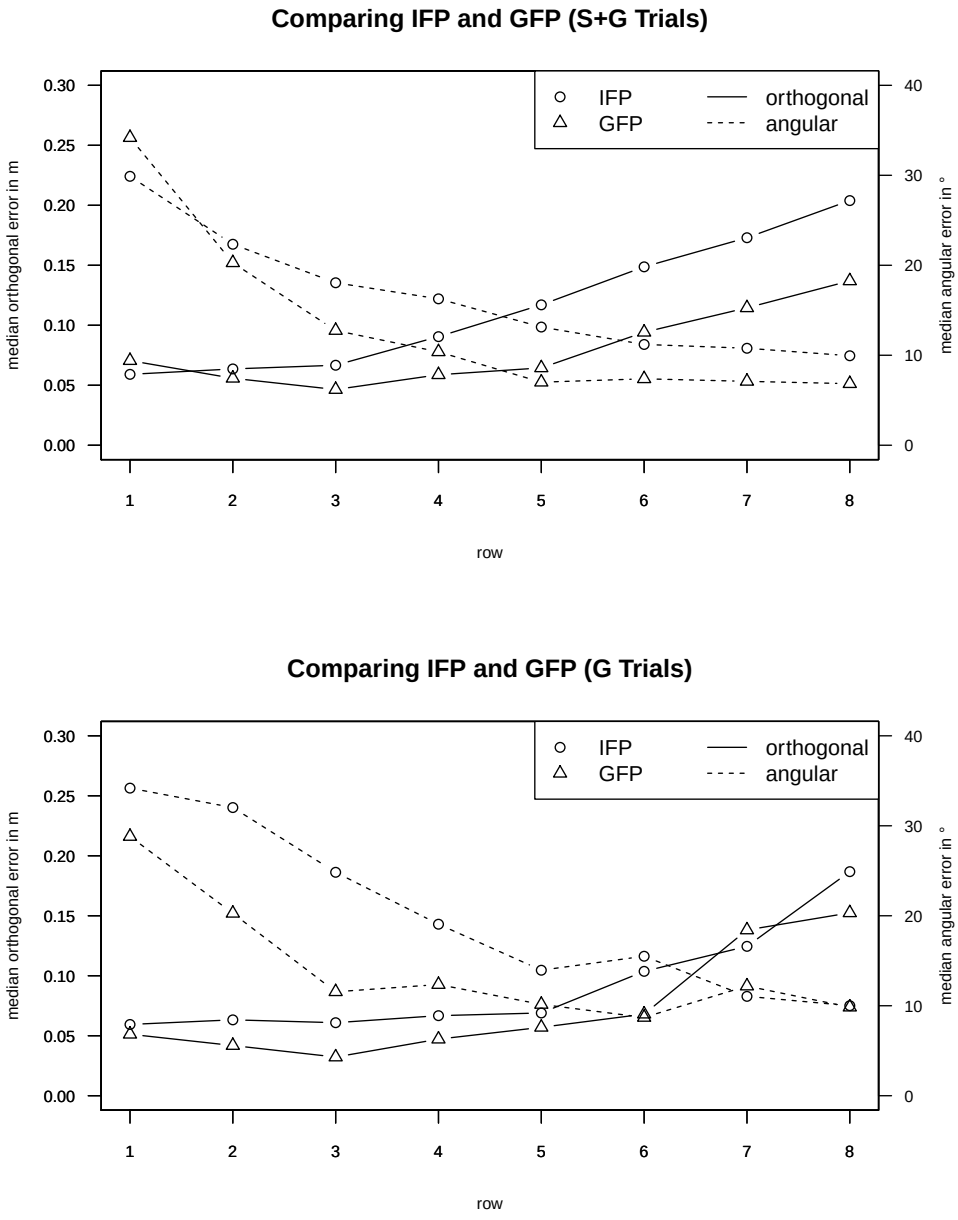


Figure 4.13: *Comparison between Index-Finger Pointing and Gaze-Finger Pointing in the S+G trials (top) and the G trials (bottom). In contrast to Table 4.2, the graphs show the medians of the errors.*

4.9.3 Patterns of Pointing Use

Gesture Handedness Of all pointing gestures produced by the right-handed DG, 76% were right-handed and 24% left-handed. As expected, most of these left-handed pointing gestures occured when pointing to the left side of the pointing domain. In column 1 to the left of the description giver, 48% of the pointing gestures were left-handed, and about 30% in column 2. On the right side of the domain, the percentage of left-handed pointing gestures was about 5% over both columns. This schema was consistent in the S+G and the G trials. Thus if not explicitly forced to point single-handedly, DGs made use of both hands, depending on the laterality of the targets, but with a strong bias towards the dominant hand.

Stroke Durations Over all games in the G trials, the minimal recorded duration of a stroke was 40 ms, which is the lowest detectable duration due to the sampling rate of 25 Hz. The longest recorded duration of a stroke was 6.04 s. The mean was 1.217 s and the median 1.28 s. In the S+G trials, the minimal recorded duration of a stroke was also 40 ms. The longest recorded duration of a stroke was 6.64 s. The mean was 1.25 s and the median 1.16 ms.

Interplay of speech and gesture The relation between speech and pointing gestures is not in the direct focus of this thesis. However, these interactions have been analyzed and this brought up some interesting findings. First of all, it can be questioned whether the DG is aware of the loss in discriminating power of his manual pointing gesture (see Figure 4.11). It is difficult to answer this question in retrospect, as the DGs were not interviewed on this specific aspect. However, there are strong indices that this is indeed the case. Take, for example, the number of words used in the speech part of the deictic expression (see Figure 4.14). Between rows 3 and 6 there is a notable increase from a mean of 3 to a mean of 6 words (the matching digits are a coincidence) that can be interpreted as a verbal compensation for the loss in manual pointing precision. The absence of identification failures by the OI in the S+G trials underlines that this is part of a successful strategy. Note that the location of the threshold between rows 3 and 6 for the number of words matches nicely with the location of the threshold between rows 4 and 5 that has been identified for the G trials as the distance beyond which the identification failures of the OI increase.

Before continuing the presentation of the results, a brief description of a new kind of visualization technique will be given, that has been developed as part

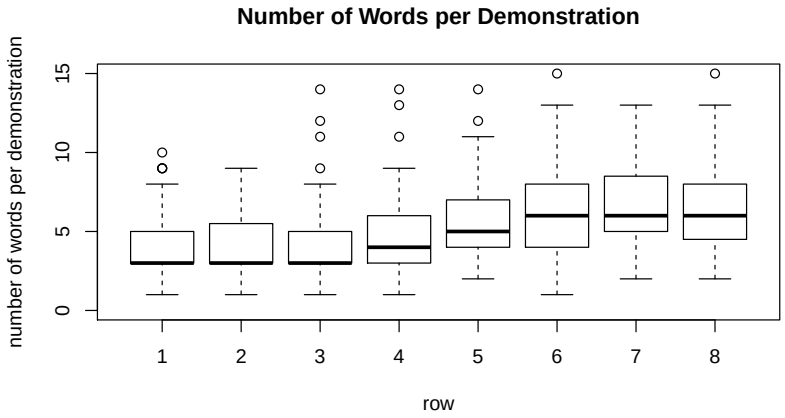


Figure 4.14: *The number of words used by the DG in his deictic reference move is depicted here as a function of the row of the referent object. Between rows 3 and 6 there is a notable increase in the number of words used, from a mean of 3 to a mean of 6 words per deictic reference.*

of this thesis to visualize certain aspects, such as the locations of index-fingers during strokes for large numbers of participants.

4.10 Visualizing Gesture Space in 3D

McNeill's gesture space has already been introduced in Section 2.3.1. The gesture space is conceived as a shallow disk in front of each interlocutor in which gestures are performed (McNeill, 1992). A schematic 2D drawing depicting his categorization of different areas in gesture space is shown in Figure 2.9. McNeill and others used this diagram to annotate gestures by marking points with a dot corresponding to the positions of relevant body parts during the gesture. Using a 2D categorization to specify gestures in 3D space is symptomatic for research on gestures and presumably constrained by the technology available. The observed interlocutors are usually recorded during the studies using a video camera from a frontal perspective, and the manual annotations are carried out post-hoc. One way of proceeding is to overlay the 2D gesture space category scheme on top of the video and note down the areas of the gesture space traversed by a specific gesture, or the area the stroke of the gesture is performed in.

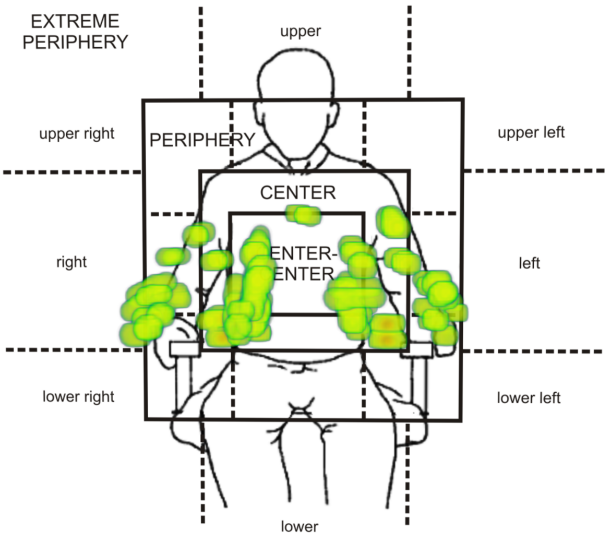


Figure 4.15: *This visualization of the gesture space has been created automatically based on the tracking data from the motion capturing system. Each blob marks an end position during a pointing stroke. If several blobs overlap, the shading is adjusted accordingly, transitioning from green (single instance) to red (multiple instances).*

The new methodology developed for the study allows for an automatic tracking of the interlocutors' gestures, and the position of the hand in the gesture space can be sampled automatically without the need for manual annotation. The picture of the gesture space in the tradition of McNeill is presented in Figure 4.15, with one difference: the frequency of gestures sampled at the same position is visualized using different shades of color, similar to the heatmaps (see Section 2.4.6 on page 29) used to visualize attention. However, to the knowledge of the author, the literature on gesture research does not provide a common way of visualizing 3D data sets on gesture production. In the following, a visualization technique for the gesture space is presented, that aims to fill this gap in displaying a real 3D gesture space.

4.10.1 Gesture Space Volumes

A visualization of the 3D gesture space should be able to show the distribution of the gesture movements in space, aggregating, for example, over time or over participants. Based on the set of 3D positions sampled by the tracking system, the frequency of gesture occurrences at each point in space can be computed,

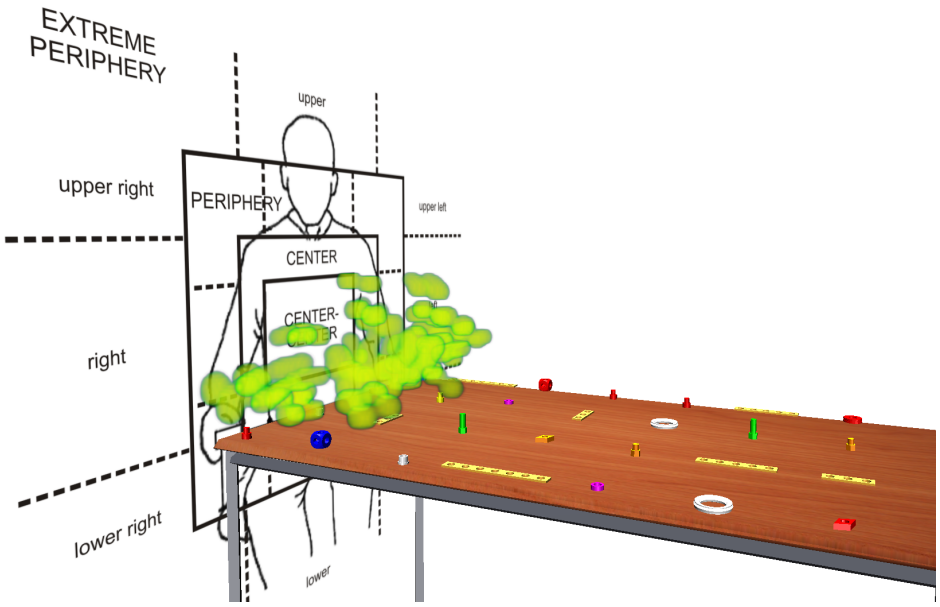


Figure 4.16: *Gesture Space Volumes visualize the 3D gesture space and thus extend the well-known gesture space visualizations of McNeill to 3D. In the interactive viewer, the perspective from which the Gesture Space Volumes are shown can be dynamically changed, and the user can zoom in on details.*

and the relative probability for a gesture occurring at each point in space can be derived. This 3D volume of frequencies or probabilities (if normalized) represents the distribution of gestures in this gesture space. In computer graphics, volume rendering techniques have been developed to visualize these kinds of data. Examples of volume rendering commonly known are found in brain imaging or flow patterns, e.g. in aerodynamics.

The visualization for 3D gesture space proposed here is the *gesture space volume* (see Figure 4.16). The gesture space volume uses volume rendering to visualize the distribution of certain aspects of gestures of one or more interlocutors in space. The original data of gesture space volumes are 3D positions of relevant extremities as sampled by the tracking system. These samples can be filtered in accordance with the aim of the visualization, for example samples during relevant gestures can be extracted. Each sample will then be associated with a 3D function that maps the sample to the volume space. This 3D function determines shape and coloring of the final visualization of the sample. An example for the manual pointing study will be provided soon. Typically, these 3D functions model a distribution. The 3D functions are then discretized on a 3D array and rendered using volume

rendering. During the rendering process, a transfer function can be applied to map the values at the individual discretized grid points to colors. This can be used to emphasize different aspects of the data.

4.10.2 Gesture Space Volumes of Manual Pointing

The gesture space volumes were developed by the author during the analysis of the manual pointing study to gain an overall picture of the pointing behavior of an individual or a group. Examples with data from the study are depicted in Figure 4.17 and Figure 4.18. For individuals, the sampled positions of the index finger are depicted for both hands during a stroke. The time span of the stroke has been manually annotated. Each sampled position is represented by a 3D function modeling a gaussian sphere (see Equation 4.1) with a radius of σ, which could represent the precision of the tracking system:

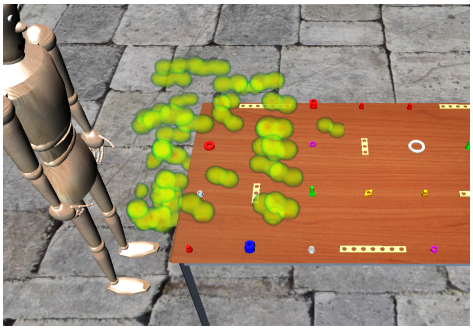
$$f(x,y,z) = de^{-\frac{(x-pos_x)^2+(y-pos_y)^2+(z-pos_z)^2}{\sigma}} \tag{4.1}$$

In the present study, the tracking system's precision was below one millimeter, which would render the visualizations nearly invisible in print, so σ was increased to 5 cm to produce more expressive visualizations. The gaussian function is also amplified by the duration the position was held, so time is represented as color in the graphics. For visualization, all 3D functions are discretized on a 3D array. The transfer function used for these visualizations is a heatmap that associates a spectrum from red over green to dark with decreasing probabilities. In addition, transparency increases with decreasing probabilities, otherwise the images would only show black boxes. Thus, a light green color represents areas that were touched during a pointing stroke, and red colors represent areas that either have been passed through by multiple strokes, which is unlikely in the intra-individual case given the tasks in the study, or where the individual dwelled for a longer period during a stroke.

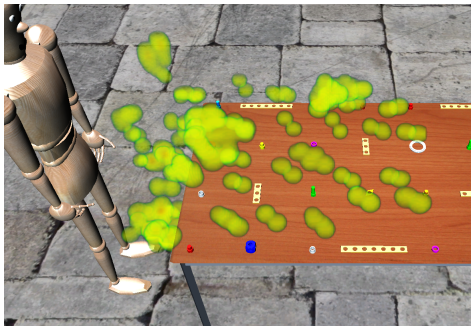
The different strategies of the description givers for coping with the situation that they are not allowed to use speech are carved out of the raw tracking data using the gesture space volumes. Person 04 is an exponent of the *leaning forward* strategy (see Figure 4.17 a,b). During the S+G trials, the pointing gestures do not exceed the second row. However, in the gesture-only trials P04 tries to decrease the distance between finger tip and referent by extending the range up to the fifth row. In the data set of interactions between 23 description givers and object identifiers, 61% of description givers follow this strategy. This explains why the orthogonal error using IFP is significantly

greater in the S+G trials: the participants use leaning forward in the G trials to reduce the distance to the referent object, and thus errors in pointing direction have a lower amplitude.

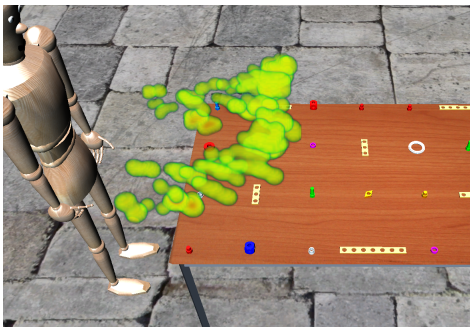
Person 07 follows a different strategy, *raising high* (see Figure 4.17 c,d), by raising the pointing hand higher above the table. The range is only slightly extended from second row to third row. This strategy is used by 48% of the description givers. The strategies *raising high* and *leaning forward* are not used exclusively, as the percentages reveal. About 30% of the description givers combine both strategies, such as for example Person 11 (see Figure 4.17 e,f). Other strategies that can be found are *frantic hand-waving* (see Figure 4.18 a,b), which happens once (4% of the cases) and an *increased dwell time* during the stroke, which happens twice (see Figure 4.18 c,d). Only three description givers do not show any different behavior in their gesture-only trials. Overall, taking the data from all participants together, the gesture space volume for the gesture-only trials is extended both along the rows of objects on the table and high above the table (see Figure 4.18 e,f).



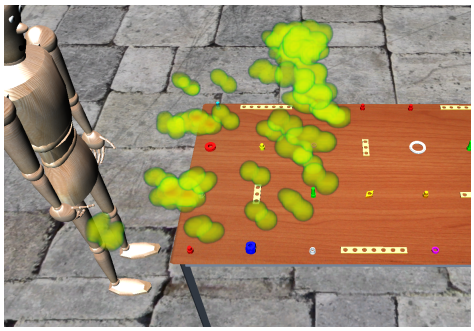
(a) P04: S+G trial



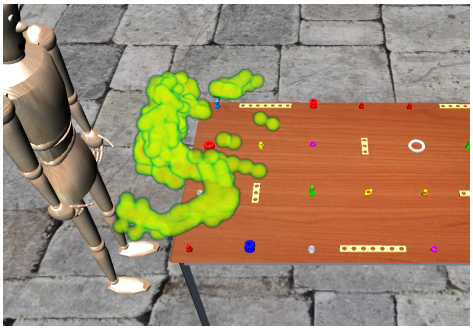
(b) P04: G trial, strategy: leaning forward



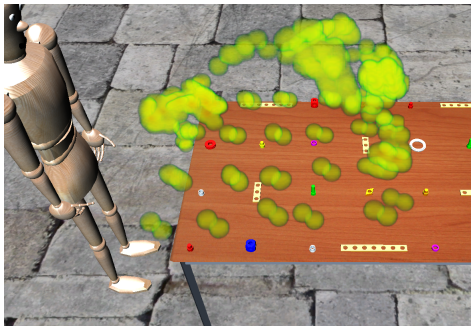
(c) P07: S+G trial



(d) P07: G trial, strategy: raising high

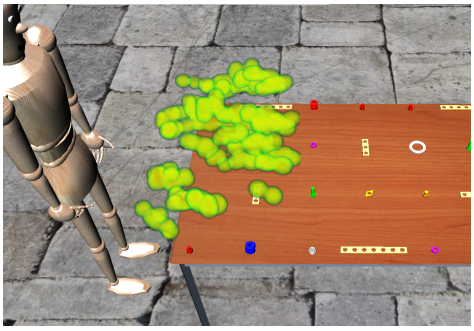


(e) P11: S+G trial

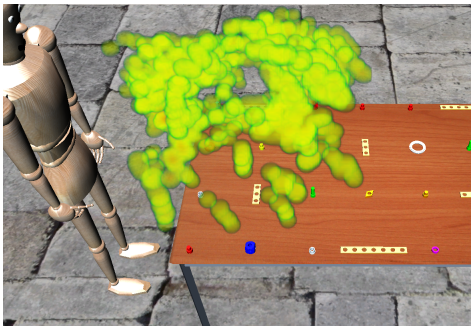


(f) P11: G trial, strategy: raising high and leaning forward

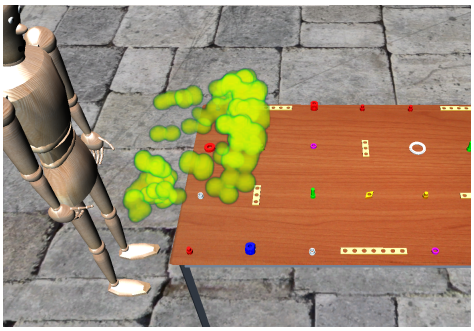
Figure 4.17: *A study on different pointing strategies using gesture space volumes. The graphics show examples from the speech and gesture trials on the left and from the gesture-only trials on the right. The wooden mannequin represents the position of the DG.*



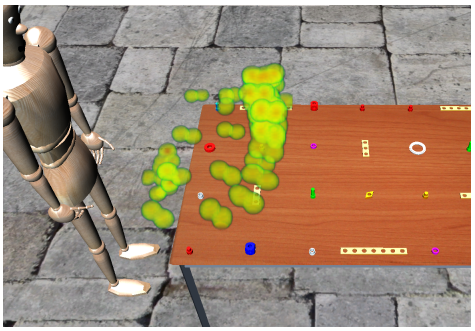
(a) P18: S+G trial



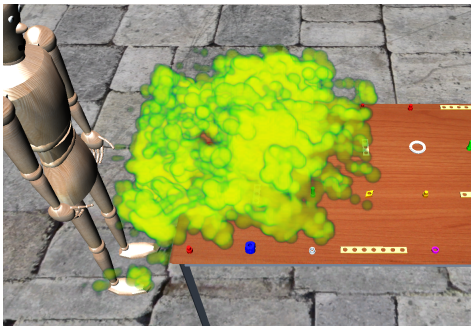
(b) P18: G trial, strategy: frantic hand-waving



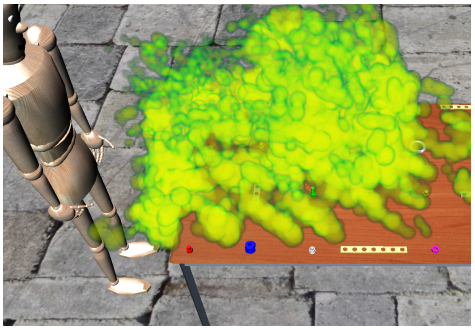
(c) P31: S+G trial



(d) P31: G trial, strategy: increased dwelling



(e) ALL: S+G trials



(f) ALL: G trials

Figure 4.18: *There are different strategies for coping with the gesture-only trials. Some lean forward to bring their index finger closer to the referent, some raise their hands, some do both and some increase their dwelling time, which is depicted as a darker red shading. Overall, the gesture space volume expands in the gesture-only trials when compared to the speech and gesture trials.*

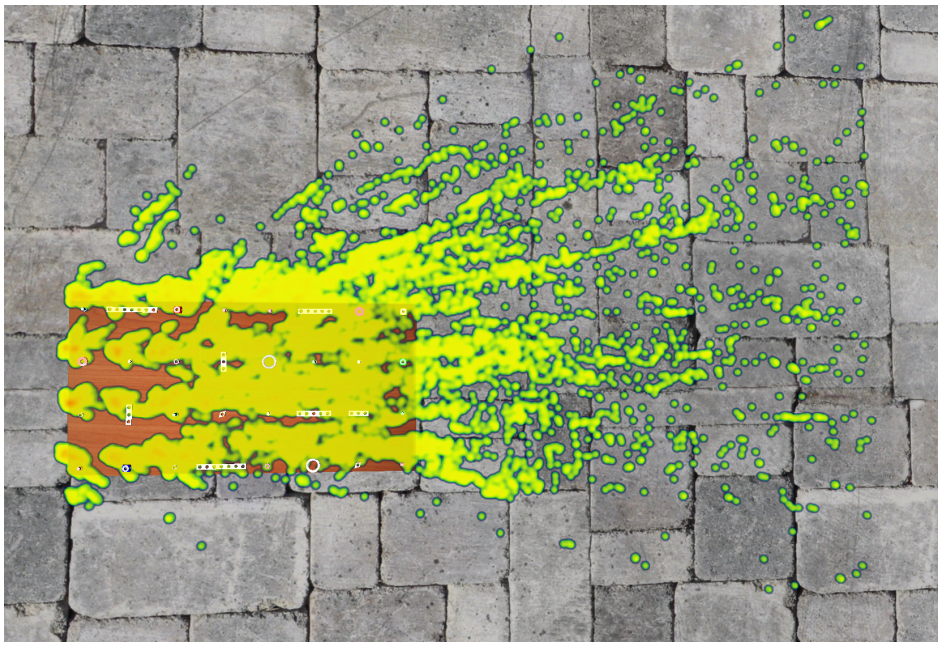
4.11 Visualizing Reference Volumes for Manual Pointing

In Section 2.4.6 on visualizing attention, heatmaps were introduced as a way to visualize the distribution of attention on a 2D image, such as a webpage. Heatmaps depict both the location (position) as well as the duration (color) of attention, typically visual attention recorded using eye tracking.

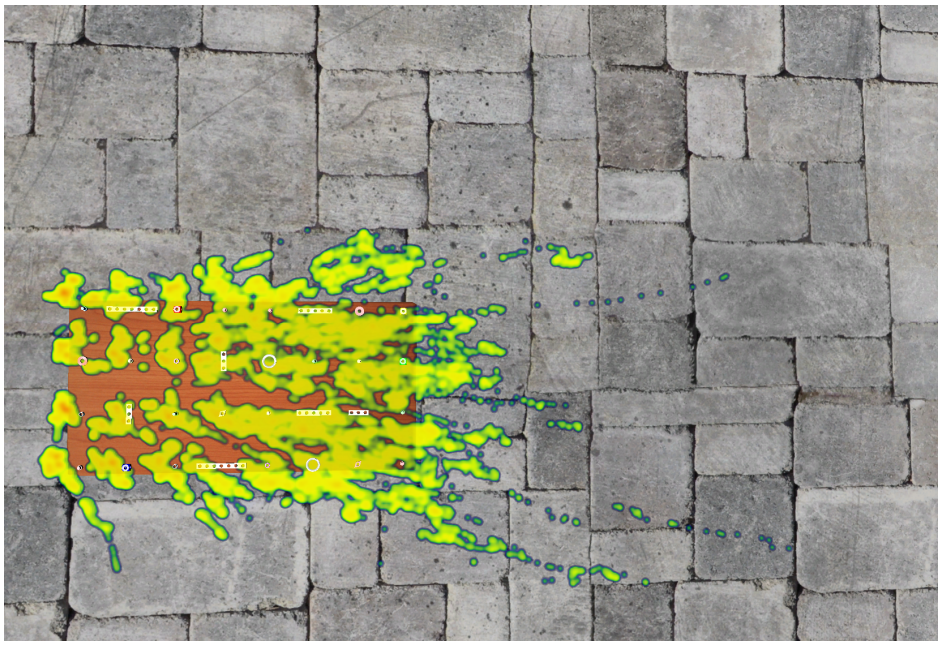
This concept can be transferred to 3D space using the same technique as for the gesture space volumes to visualize the locations that have been the target of a pointing gesture, defining the *reference volume*. The reference volume is the space that has been the target of a referring act, which is a cautious expression to emphasize that the reference does not necessarily need to be successful nor does the referring act necessarily need to be intentionally directed to this specific space. Whether the referring act itself must be given intentionally or not is arguable. In principle, any produced utterance or body movement that has the outer form of a referring act can contribute to a reference volume. It depends mainly on the perspective. The DGs, for example, might only include reference acts in their reference volumes if they produced them intentionally. Also, they will probably interpret their own pointing gestures as being directed exactly at the target referent. The OIs, however, can only add reference acts they detected as such.

For the present study, this means that the intersections of the pointing rays with the surface of the table can be taken as points being technically referred to, although the intention of the DG had been to refer to the object. This was done by applying the IFP and the GFP model for vector extrapolation on both trials. Again a gaussian function was used to model individual points, here with a σ of 1 cm. In the S+G trials, see Figure 4.19, the difference between GFP and IFP can clearly be seen. GFP leads to a more compact reference volume, while IFP leads to a reference volume projecting far over the border of the domain.

In the G trials, see Figure 4.20, the difference between GFP and IFP is less clear, which is according to expectations. However, for GFP the reference clouds surrounding the objects in the first four rows stand out more clearly than those for IFP.

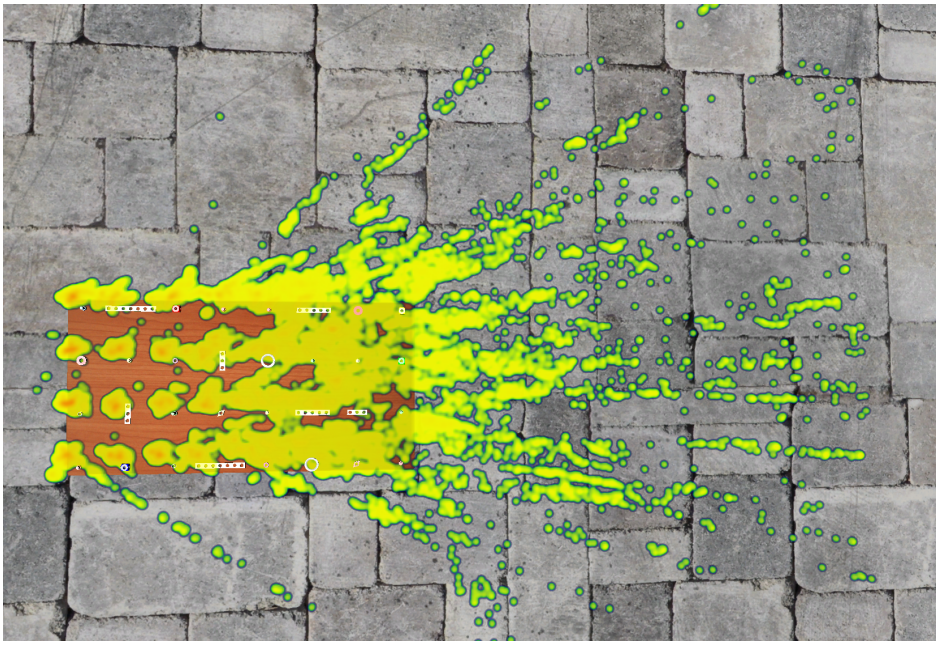


(a) IFP in S+G trials

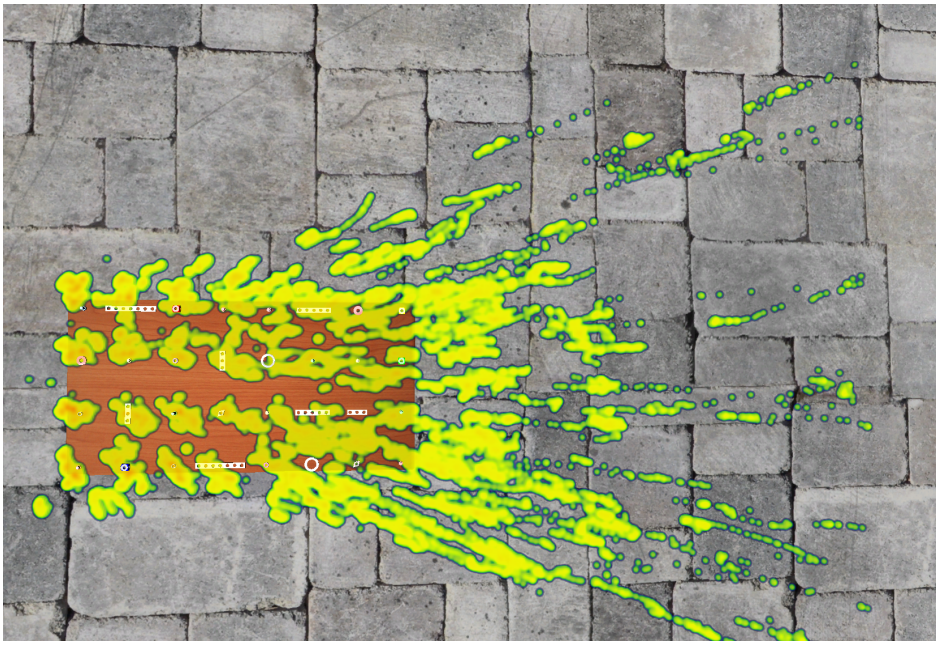


(b) GFP in S+G trials

Figure 4.19: *Reference volumes for the S+G trials. GFP leads to more compact reference volumes, while the reference volume of IFP extends far beyond the border of the domain of possible referents.*



(a) IFP in G trials



(b) GFP in G trials

Figure 4.20: *Reference volumes for the G trials. The difference between GFP and IFP is less clear than for the S+G trials.*

4.12 Summary

This chapter has presented an extensive study on multimodal pointing to assess manual pointing gestures in an identification game between two participants. Several methodological challenges had to be faced when approaching this study. Finally, the study was successfully conducted using an empirical methodology based on a combination of well-tried linguistic methods with state-of-the-art tracking and virtual reality technology. To this ends, the I*nteractive* A*ugmented* D*ata* E*xplorer* (IADE) (see Section 4.6 and Section 4.8) has been developed, which provides the technical basis for the recording and integration of multimodal data. I*nteractive* A*ugmented* D*ata* E*xplorer* (IADE) offers exciting new possibilities for data-driven computer simulation. With its basis in virtual reality technology, IADE also allows the researchers to literally immerse into their data and inspect the body movements from any perspective.

A first conclusion of the study is that manual pointing is fuzzier than expected. Even in simple domains, as the one used in the study, the description givers (DGs) fail to accurately direct their pointing gesture at a single object. In the G trials only the rows within direct reach of the DGs yielded good identification results, but the object identifiers (OIs) failed to identify many distant referents. In contrast, the OIs were able to identify all the referents in the S+G trials. Considering the weak identification results for the distal rows in the G trials, these successful identifications have to be based on information from other modalities than manual pointing. The increased number of words used by the DGs in deictic expressions referring to the distal rows indicates that speech is used to compensate for the inaccuracy of the pointing gesture.

Considering the *where*-question, two models for the direction of manual pointing gestures have been tested: *IFP* and *GFP*. It has been shown that GFP produces significantly lower angular errors than IFP – at least for the distal rows in the S+G trials. The course of the angular and orthogonal errors of IFP and GFP shows a clear trend towards low angular errors between 9° and 11° in the S+G trials and a linearly increasing orthogonal error. The results for the G trials show small irregularities in the distal rows, and a clear distinction between IFP and GFP cannot be made. Nevertheless, they appear to follow a similar trend approximating slightly larger angular errors between 11° and 12°. *In the mean, GFP and IFP approximate the ideal pointing direction up to an angular error between* 9° *and* 11° *in the S+G trials. GFP performs slightly but significantly better than IFP in the distal rows.*

Given that the distances between the objects in the grid were only 20 cm, the mean orthogonal errors were found to be too high to uniquely identify the referent, which consequently led to a drop in the success rate for the distal rows in the G trials. Pointing rays cast from the predicted pointing direction were actually closer to the neighboring objects (mean orthogonal deviation M>10 cm in the distal rows) than to the referent in most of the cases. *The data on the accuracy of the pointing direction predicted by the two candidate models IFP and GFP suggests that vector extrapolation is not a suitable model for the extension of pointing.*

An important finding is the dichotomization of the gesture space into a *proximal* and a *distal area*. This finding is supported by evidence from the interaction between speech and gesture, as well as the distribution of the measured errors. The border between proximal and distal area lies between the 3rd and the 5th row (47.75 cm to 87.75 cm). The exact position might correlate with the armlength of the description giver.

The pointing behavior to objects in row 8 is different from that to closer rows. The DGs use the fact that row 8 marks the border of the pointing domain and exaggerate their gestures by overshooting to clearly differentiate them from gestures towards objects in row 7.

To visualize the recorded 3D movements that occurred during the relevant gesture phases, a new visualization of gesture space in 3D, the *Gesture Space Volumes*, has been developed. These visualizations show the positions of gestures in the gesture space over time, either for a single trajectory or integrating over all gestures from several interlocutors. The Gesture Space Volumes generated for the positions of the pointing hand during the stroke revealed different coping strategies which were used by the DGs to compensate for the deficiencies of pointing to the distal area: *leaning-forward* and *raising-high*.

These findings shed new light on the interaction between speech and gesture. In the S+G trials the DGs seemed to compensate for loss of precision of their pointing gestures when targeting distant referents by increasing the words in their deictic expressions. In the G trials, when they were not allowed to speak, they put more effort into their pointing gestures, either by raising them higher or by leaning forward to reduce the distance to the referent object. This is a bidirectional interaction between speech and gesture: either modality is adapted to compensate the absence of the other.

The corpus of manual pointing acts collected during this study will be used for data-driven modeling in Chapter 6 to find optimized models for the direction and the extension of pointing.

Chapter 5

Gaze Pointing

Whereas the previous chapter provided insights on manual pointing from a study on demonstrations to real objects, this chapter focuses on gaze pointing. Two studies are presented that investigate the use of gaze as a pointing device in human-computer interaction.

In the first study, the combination of eye tracking and motion tracking was tested in a virtual reality setting where users were asked to use gaze pointing to refer to spheres at different levels of depth. In this scenario, the participants were able to walk freely in the area of the TRI-SPACE virtual environment at the A.I. group at Bielefeld University. The study provided results on the accuracy and precision of gaze pointing using a *direction-based* approach. It is also the first study that tested the gaze-specific components of the prototype of the interaction framework DRIVE described in the Appendix (Chapter A).

The second study investigated *location-based* approaches to interpret gaze pointing. The participants were asked to use gaze pointing to refer to certain objects located in a complex assembly of Baufix parts. Using two binocular eyetracking devices, their eye movements were monitored, and the location of the point of regard in 3D was estimated based on two algorithms presented in Section 3.3.4, triangulation and the PSOM approach. Both algorithms and two different eyetracking devices were evaluated.

This chapter concludes with the presentation of a visualization technique that extends standard 2D visualization methods to 3D space.

5.1 Study 1: Direction-based Pointing

Gaze-based interaction is not part of the standard repertoire of virtual reality systems, as is the case with motion tracking systems and basic gesture-based interactions. In fact, only few desktop or head-mounted virtual reality systems make use of gaze-based interaction, and even less do so in immersive CAVE-like environments. The aim of the study presented in this section is to test a new framework for gaze-based interaction in a virtual reality setting, and especially to gather data on the accuracy and latency of the interaction. This study has been published in Pfeiffer (2008). In contrast to the set-up used in the previous study on manual pointing, the target objects used in this study are virtual objects.

In preparation of this study, a prototype of the gaze-based interaction framework was developed. This prototype is described in some detail in Section 5.2.1. An essential part of the framework is the mapping from the 2D gaze positions provided by the vendor-specific eyetracking software to the orientation of the eyes of the participant in the 3D world coordinate system. This mapping requires a calibration process which can easily be handled by the user, is operated self-controlled and does not require too much time. A calibration process that complies to these requirements is presented in the same Section 5.2.1 as well.

As one of the questions is the latency that can be achieved using this set-up, the specification of the hardware that was used for the study is given in Section 5.2. One problem was to find a procedure to measure the latency of such an interactive system. A task-specific solution is proposed in Section 5.3 with the Visual Ping procedure.

The study itself is presented subsequently. The questions addressed by the study are:

- What latency can be achieved with the system? Is it fast enough to be used in interactive settings?
- How accurately can the 2D gaze positions be mapped to 3D space?

5.2 Study 1: Hardware Set-Up

In the following, the hardware set-up of the TRI-SPACE virtual environment in the AI Lab at Bielefeld University in 2007 is described (see Figure 5.1).

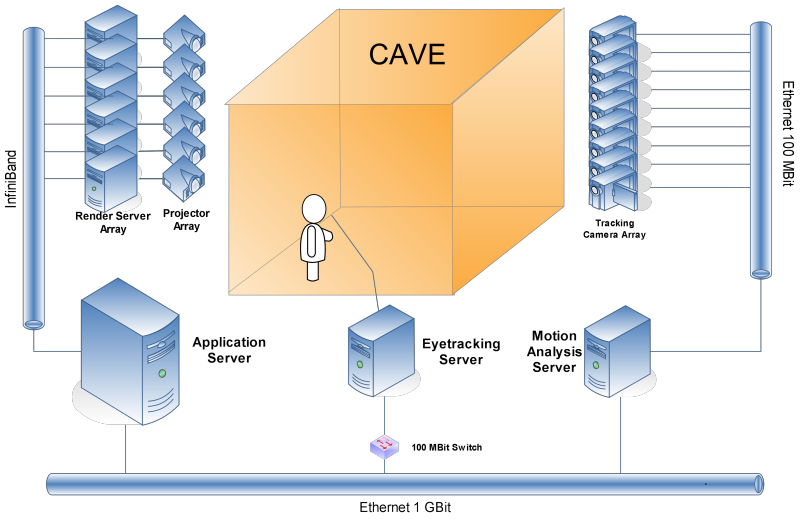


Figure 5.1: *The immersive virtual reality set-up used for the study on direction-based gaze pointing in 2007. Compared to the study on manual pointing, an eyetracking server has been added to the set-up, as well as a GBit Ethernet connection between the interaction servers.*

The virtual reality application was driven by AVANGO (Tramberend, 2001) on a dual AMD Opteron 248 2.2GHz machine with 3GB RAM. The views were distributed by Chromium (Humphreys, Houston, Ng, Frank, Ahern, Kirchner & Klosowski, 2002) (version 1.6) to six render clients, each with AMD Athlon 64 3000+, 1GB RAM, and a NVIDIA Quadro FX 5600 card. They were running Ubuntu with a Linux Kernel 2.6.20 and a NVIDIA Kernel Module version 100.14.19. The cluster was networked by InfiniBand using Mellanox Technologies MT25204.

A ViewPoint PC-60 EyeFrame BS007 eye tracker (see Figure 5.2(b)) manufactured by Arrington Research Inc. (2008) was used, which could be easily combined with the markers for the optical tracking system ARTtrack1 by Advanced Realtime Tracking GmbH (2010), and the polarized filters for the stereo projection. The ViewPoint PC-60 offers moderate resolutions in time and space (see Table 5.2(a)), which should be adequate for normal interaction tasks (disregarding saccades or microsaccades). Of the two different operation modes, higher temporal resolution was chosen over higher spatial resolution, as a higher priority was given to the reaction time of the system. The eye tracker was driven by the ViewPoint software in version 2.8.3,33 on an Intel Core2Duo 6600 machine with 2.4 GHz running Windows XP Professional SP 2. The machine was connected to the virtual reality application via a 100

	ViewPoint PC-60
temporal res. (Hz)	30 / **60**
optical res. (pixel)	640×480 / **320×240**
accuracy	0,25° - 1,0°
precision	0,15°

(a) Technical specifications of the ViewPoint PC-60 eye tracker from Arrington Research

(b) Eye tracker mounted with markers for optical tracking

Figure 5.2: *In order to reduce latency, the eye tracker was configured to run at 60Hz with a lower optical resolution in the study. The eye frame of the eye tracker has been mounted with a 6DOF marker from the optical tracking system to monitor the position and orientation of the tracker. Polarized glasses were also added, as required by the stereo projection technique used by the TRI-SPACE.*

MBit Ethernet connection (see Figure 5.3(a)). Tracking the user's head was done using a 6DOF marker to the left side of the frame of the ViewPoint PC-60 (see Figure 5.2(b)). The tracking set-up consisted of nine ARTrack1 cameras which ensured a stable presence of the 6DOF marker in at least three cameras during the interactions. The tracking data was sent to the application computer over a 100 MBit Ethernet connection (see Figure 5.3(a)).

5.2.1 Study 1: Software Framework

A first prototype of the gaze interaction framework DRIVE (see Chapter A) was created and tested in this study. The basic architecture is based on a data-flow network partly embedded in the scenegraph of the virtual reality application (see Figure 5.3(b)). The *EyeNode* represents one eye of the user in the scenegraph of the virtual reality application. It is fed by the tracking data provided by the head tracking (*Head actuator*) and the eye tracking (*Eye Data Client*). Based on the current tracking data and information gathered using a calibration procedure, the *EyeNode* constructs a position and an orientation vector representing the pathway of the eye's visual axis in space. The calibration is handled by a *Calibration Module* which manages a step-by-step procedure using a regular grid of 3D spheres to gather fixations

to reference points in the relevant view area. By letting the *Calibration Module* follow head tracking in real-time, the calibration grid maintains a stable position relative to the center of the eye. Using this closed system approach to calibration, eye tracking can be calibrated independently of the viewing perspective of the head.

In this prototype, object selection is realized using a pointing-cone model with an aperture (the angle at the apex) of 5°. This is motivated by the range of high acuity vision, as described in Section 2.4.1 on visual attention. In the *Ray Construction* the central axis of the cone is calculated based on eye position and orientation. A *Histogram* node collects the angular distances of all objects within an angle of 2.5° around the ray during an interval of 400 ms. From this histogram, the object with the highest ranking for at least 200 ms is taken as the fixated object. The minimum duration of 200 ms is motivated by the findings described in Section 2.4.4 on the timing and duration of fixations. The filtering is necessary to remove noise, for example if the eye tracker shortly lost the eye, which happens during blinks.

It has to be noted that a non-standard interpretation of fixation is used here. The fixation model that is used could be coined *semantic fixation model*, as it is based on stable fixations of individual objects, not on stable positions or directions of the eye, as is usually the case. In general operation, the eye tracking software has no knowledge about the semantic structuring of the visual field, and thus the software calculates fixations based on eye movements alone. For interactive settings, where the user is able to move around freely, the object-oriented fixation model subsumes the standard fixations as well as small eye movements called smooth pursuit that happen when the eye relocates the focus of attention if either the object or the human has moved. This leads to a more robust detection of fixations in dynamic settings.

The semantic fixation model only accounts for fixations to objects as a whole, not on substructures. Yet the objects in this study are small and show no interesting substructures. Other approaches exist that also detect fixations on substructures. Duchowski et al. (2002) used a more fine-grained fixation model based on gaze intersection points (GIP). A GIP is calculated by intersecting the gaze ray with the geometry of the object, and fixations can be detected on the level of individual triangles. The fixation model used in the current study, however, is bound to objects, which are the atomic interactive entities of the scenario. Longer and/or frequent fixations of objects could then result in a selection, based on an application-specific threshold (see again Section 2.4.4). However, in the current study, the gaze pointing was done iteratively and out of context, so the minimum fixation duration of 200 ms was used as an

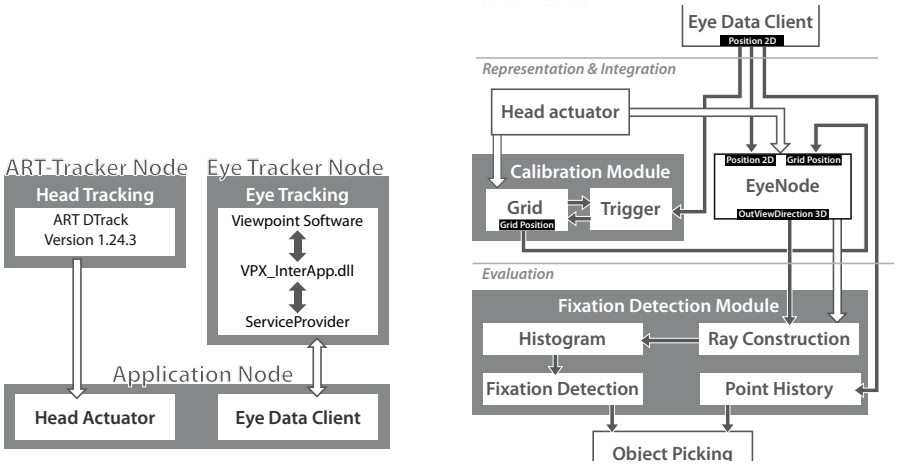


Figure 5.3: *(a) The interaction handling is distributed over the network. The ART tracking system and the Arrington Research eye tracker are controlled with separate computers. The tracking data is then sent to the application over network using a proprietary protocol. (b) Embedded into the application the tracking data is interpreted using a data-flow graph. Details are given in the text.*

indicator of a gaze pointing act instead of the longer dwell times which are required in more versatile settings.

5.3 Study 1: Visual Ping

To evaluate accuracy and latency of the system, a human-in-the-loop procedure was developed for the study called *visual ping*. In this procedure, the task of the participants is to fixate a single highlighted sphere from a test-grid of 64 spheres. These spheres are placed on one of four test-grids in a plane perpendicular to head orientation at the distances near (0.7 m), normal (1.7 m), far (2.7 m), and very far (6.7 m). All test-grids are placed in such a way that they are within angular eye movements of horizontally -35.29° to 35.36° and vertically -36.33° to 36.33°. The test-grids exceed the grid used for calibration in each direction by about half the distance between the rows/columns. Thus the calibrated points lay in between the points of the test-grids.

The loop of the visual ping starts with the participant fixating the highlighted sphere (all others are invisible). When the system detects this fixation, it

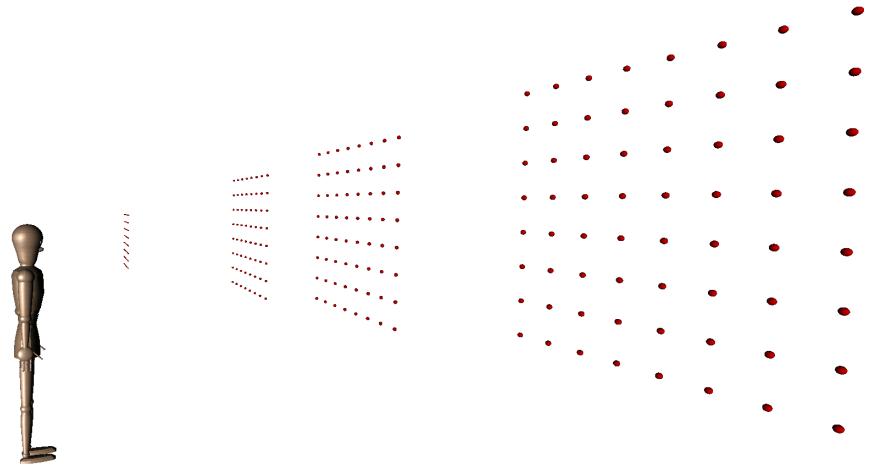


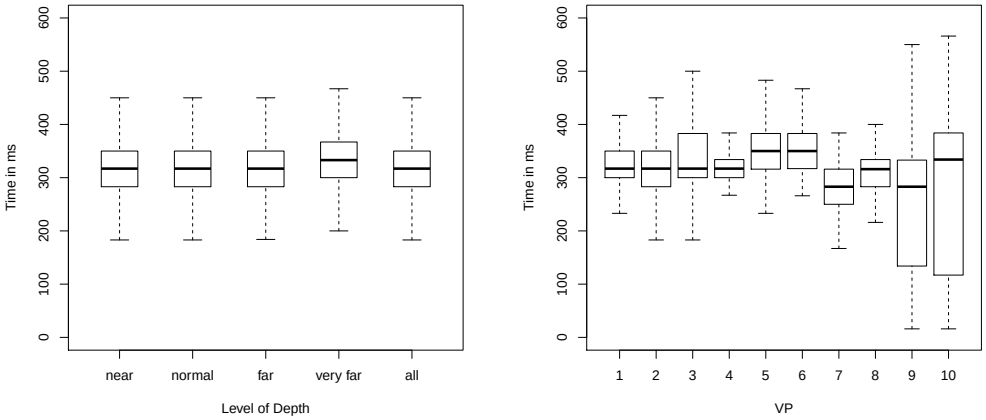
Figure 5.4: *The mannequin demonstrates the set-up of the direction-based gaze pointing study. Participants had to point to 256 target spheres via gaze, following a given sequence. The spheres were arranged in 4 quadratic grids of 64 spheres each. The grids were presented at 4 different distances.*

hides the current sphere and highlights a new sphere from the grid taken at random. This exact moment defines the start time of the visual ping. The participant detects the vanishing of the fixated sphere and answers with a search movement of the eyes for the next sphere. The search is easy, as the new sphere is the only visible object and well within the visual field. The time of the first eye movement that is detected moving more than 2.5° away from the originally fixated position is taken as the response. The difference between the time of the response and the start time of the visual ping defines the wanted *latency*. Once the participant has found and fixated the newly highlighted sphere, the procedure continues. The deviation between the position of the detected fixation and the target sphere defines the *accuracy*.

This loop is iterated over all spheres within each test-grid. In between individual grids, a calibration run with the 4x4 calibration grid is executed at normal distance.

5.4 Study 1: Results

A total of 10 untrained people with no immersive virtual reality experience participated in the study (6 women and 4 men). The mean age was M=27.7 years with a standard deviation of SD=6.17 years. The recorded data was



(a) Latencies for each of the four distances over all participants

(b) Latencies for each participant over all distances

Figure 5.5: *Overview of the results for the visual ping test.*

cleaned by removing individual fixations beyond 2 SD. These were mostly outliers with large vertical deviations that can be attributed to eye blinks. Using this procedure, altogether 10.43% of the entries were removed.

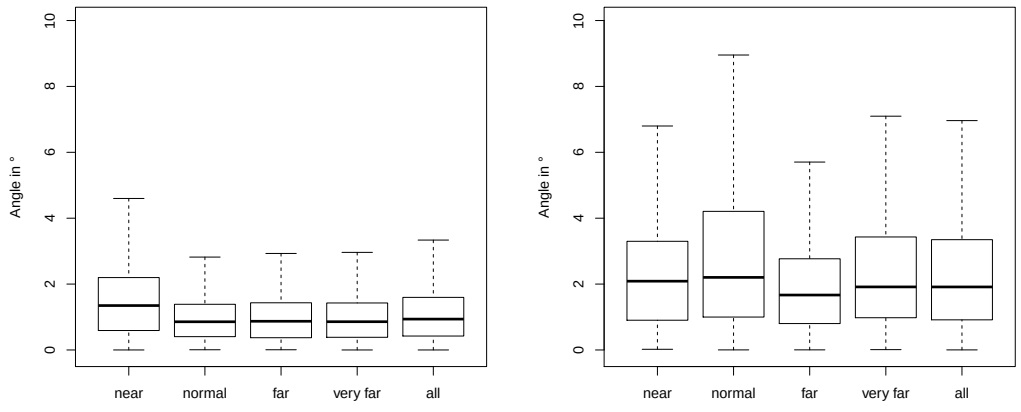
Latency The results for the latency in the visual ping procedure are depicted in Figure 5.5(a) for each distance. The mean latency over all distances is 307.9 ms, the median is 317 ms and the standard deviation is 99.9 ms. The results for the participants are depicted in Figure 5.5(b).

Accuracy The accuracy of the detected fixations is depicted in Figure 5.6(a) for the horizontal and in Figure 5.6(b) for the vertical deviation. The horizontal accuracy over all distances is 1.18° (mean) or rather 0.94° (median). The precision (in standard deviations) is 1.51°. The vertical accuracy over all distances is 2.52° (mean) or rather 1.91° (median). The precision (in standard deviations) is 2.24°. A detailed overview is given in Figure 5.7 with median and standard deviation for each grid point.

5.5 Study 1: Discussion

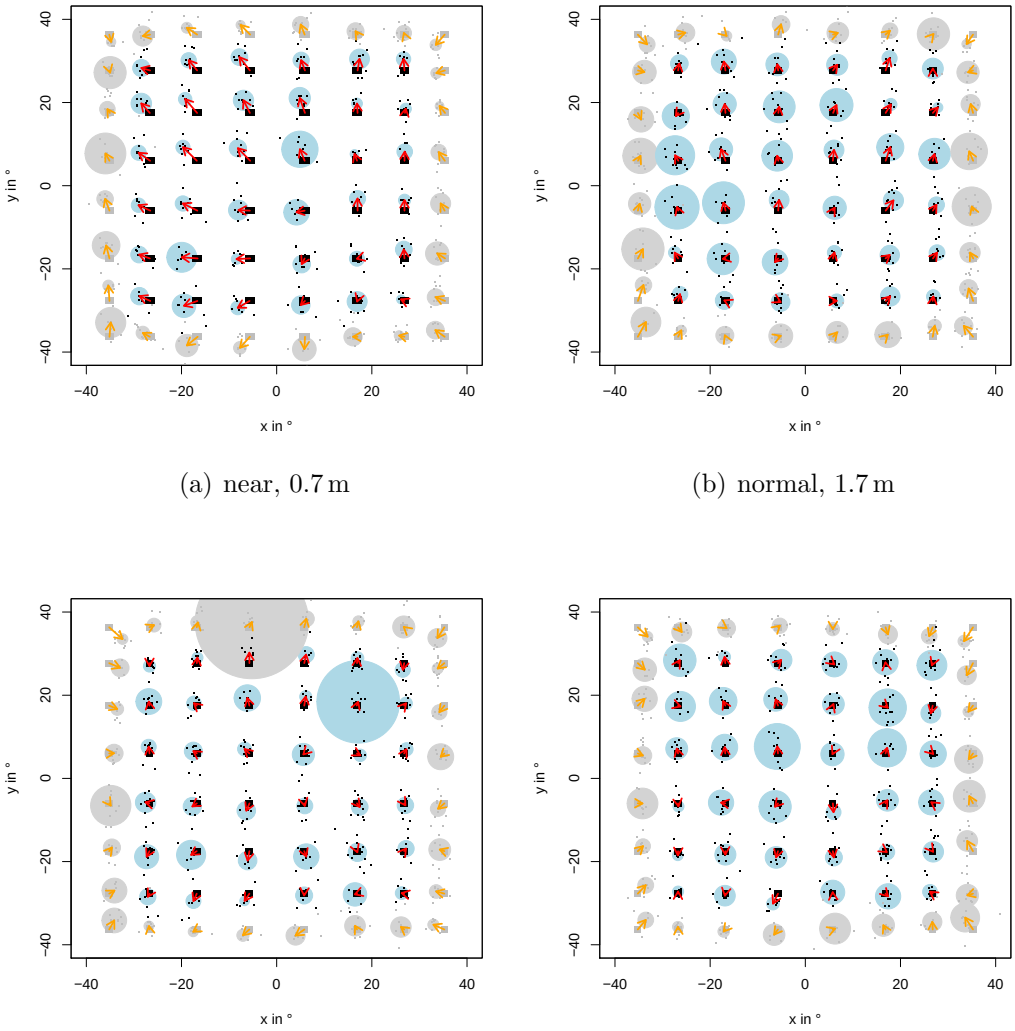
The combination of a lightweight eye tracker with an optical tracking system allows the user to interact freely in the CAVE-like system. The polarized glasses needed for the stereo projection fit well in the frame of the eye tracker's glasses. The cameras of the eye tracker, however, needed a clear sight of the eye below the frame of the glasses. Systems with indirect eye recording over a semi-transparent mirror might be more difficult to handle. No interference between the infrared optical tracking system and the infrared eye tracking was observed. However, an additional infrared LED was needed in the dark environment of the CAVE to cast enough light on the eye for a robust eye tracking.

The results from the user study exceeded expectations. An accuracy of about 1° on the horizontal axis is nearly perfect, considering that the opening angle of the foveal high-accuracy vision is about 2°. The vertical accuracy is also good, although it is less than the horizontal. This fits nicely with the findings of Chi & Lin (1997) presented in Section 2.4.2, who also detected higher errors (or larger optimal object extensions) for the vertical. In most applications the slightly larger vertical inaccuracy should not matter, as horizontal differences



(a) Horizontal accuracy (b) Vertical accuracy

Figure 5.6: *The mean horizontal accuracy of the detected fixations is quite high, with 1.18°. Vertical accuracy, however, is quite low with a mean error of 2.52°.*



(a) near, 0.7 m (b) normal, 1.7 m

(c) far, 2.7 m (d) very far, 6.7 m

Figure 5.7: *The plots show the angular accuracy of the detected fixations for each distance. The arrows show the deviation of the median, and the circles highlight one standard deviation around the median. The outer grid-points exceed the calibrated area and are depicted in a lighter color.*

are more important, for example in stereo vision. Also, if a higher accuracy is required, the eye tracking device could be operated in high resolution mode, which, however, has not been tested here. The performance was also quite stable over all tested distances, even though calibration was done only at

normal distance. This stability is not initially surprising, as the test-grids were arranged to overlap exactly. However, while the normal distance was more or less presented exactly on the projection surface and thus was not affected by ghosting (shine-through of the image presented to the other eye) or other influences due to stereo projection, the visual systems of the participants had to cope with disparity for the near, far and very far distances. Nevertheless, the fixations were detected very accurately. This projection technique therefore does not seem to reduce accuracy and one can safely rely on a single distance for calibration.

The results for latency of the visual ping show that the performance was quite similar for every tested distance. The individual performances were also quite comparable for practical reasons, except for participants 9 and 10, who showed very large differences in their latencies. The question remains, how a mean latency of 307.9 ms should be rated. For a meaningful interpretation, the performance of the human has to be separated from the overall system performance. The model for human performance of Card, Moran & Newell (1983) might provide a rough approximation for this separation. According to their model, the perception of the missing sphere (100 ms), the deliberation about the task (70 ms), and the issuing of the motoric response (70 ms) should sum up to 240 ms. The contribution of the system to the latency would then be about 70 ms, including frame grabbing with 60 Hz, image processing, networking and visualization with a frame-rate of 60 Hz. Such a latency could be noticeable if contingent continuous interaction is required. If, for example, the point of regard is visualized in real-time, the latency would make the visualization trail the actual fixation. However, for discrete interactions such as gaze pointing the latency appears to be fair enough.

5.6 Study 2: Location-based Pointing

The first study on gaze pointing focused on direction-based pointing (see also Section 3.3.1), which is in its principles similar to the vector extrapolation used in the study on manual pointing. An alternative to the direction-based pointing is the location-based pointing that is possible with gaze (see also Section 3.3.4). In principle, location-based pointing should be superior to direction-based pointing, as it rigorously restricts the referential space. Thus, the questions driving this study are:

- Is it possible to estimate the point of regard in 3D space?
- How do the different algorithms presented in Section 3.3.4, triangulation and PSOM, perform in terms of accuracy?
- Does it need a high-end eye tracker, such as the SMI EyeLink I, or is a medium-sized device, such as the Arrington ViewPoint PC60, sufficient?
- Is it possible to point to objects via gaze using an estimated point of regard in 3D space?
- Does location-based pointing have advantages compared to direction-based approaches in practice?

Parts of this study were conducted by Matthias Donner as part of his diploma thesis, which was supervised by the author. The results were published in Pfeiffer, Latoschik & Wachsmuth (2009) and presented in Pfeiffer, Donner, Latoschik & Wachsmuth (2007a) and – awarded third place in the best paper and presentation competition – in Pfeiffer, Donner, Latoschik & Wachsmuth (2007b). This work was partly funded by the German Research Foundation within the Collaborative Research Center 673 *Alignment in Communication*, and by the EU within the project PASION (Psychologically Augmented Social Interaction Over Networks). The following presentation of the study is an extended version of Pfeiffer et al. (2009).

5.7 Study 2: Hypotheses

The questions formulated above were investigated in terms of the following three hypotheses:

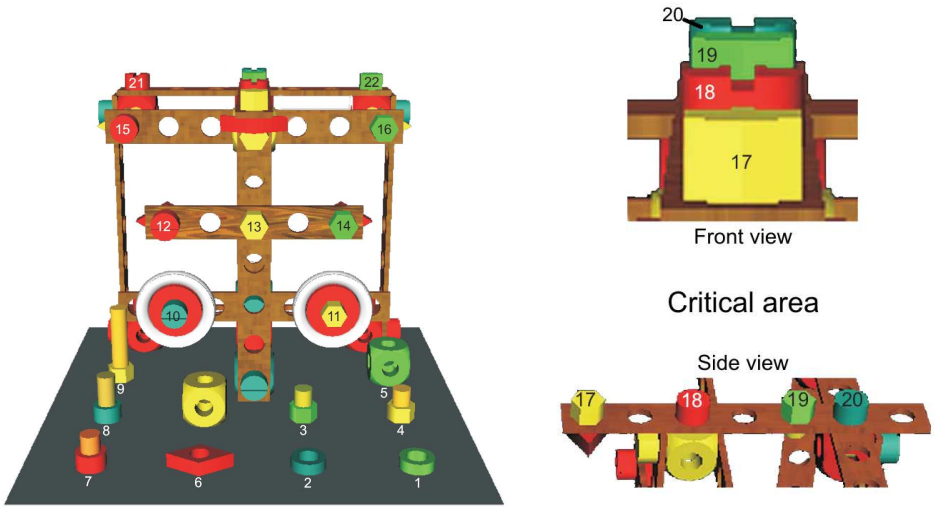
Table 5.1: Technical details of the two eye tracking systems tested

Features	Arrington PC60	SMI EyeLink I
temporal resolution (Hz)	30 / 60	250
optical resolution (pixel)	640×480/ 320×240	-
deviation from real eye position	0.25° - 1.0° visual angle	$< 1.0^\circ$ visual angle
accuracy	0.15° visual angle	0.01° visual angle
compensation of head movement	not possible	$\pm 30^\circ$ horizontal, $\pm 20^\circ$ vertical

A: The PSOM-approach is more precise and accurate than the triangulation-approach Of the two algorithms, the PSOM should have noticeable advantages. The PSOM adapts to individual biases in the alignment of the visual axes of the eyes and can compensate minor errors of the 2D calibration. This approach is therefore expected to provide higher precision and accuracy when compared to triangulation.

B: The SMI EyeLink I provides higher precision and accuracy in this task than the Arrington ViewPoint PC60 Two different head-mounted eye tracking systems were tested (see Figure 3.5, left and right): the EyeLink I from SMI as a representative of high-end devices ($> €\,30{,}000$) and the ViewPoint PC60 from Arrington Research as a representative of medium-scale devices ($< €\,12{,}000$). The technical details presented in Table 5.1 show that the device offered by SMI has noticeable advantages regarding temporal resolution (more than 4 times faster than the ViewPoint PC60) and accuracy (one fifteenth of the error reported for the ViewPoint PC60).

C: Gaze pointing to partly occluded objects can be disambiguated using the 3D point of regard Exploiting knowledge about the depth of a fixation should improve the disambiguation of difficult cases where objects are partially occluded, but have significant differences in depth (see the critical area in Figure 5.8). Therefore this approach should have a higher success rate for gaze pointing to such objects than traditional direction-based approaches.



(a) In the online version you may click on this image to explore a 3D view of the setting.

(b) This set of occluding objects defines the critical area for the 3D selction algorithm.

Figure 5.8: *Position of the objects in the model (left). The objects 17 to 20 define the critical area where direction-based pointing leads to ambiguities (right).*

5.8 Study 2: Scenario

In the study the participants looked at a 3D scene showing a structure built from Baufix toy building blocks (see Figure 5.8). The dimensions of the relevant target objects are provided in Table 5.2. A 21" Samsung SyncMaster 1100 cathode-ray monitor was used together with a NVidia Quadro4 980 XGL, and Elsa Retaliator consumer class shutter-glasses for the stereoscopic projection. Both eye tracking systems were prepared to be used in monitor-based settings. The implementation of the experiment was based on the 3D extension of the VDesigner software that was developed by the author for a previous study, published in Flitter, Pfeiffer & Rickheit (2006).

The study had four conditions, resulting from an intra-personal covariation of two tested eye trackers and two algorithms. To stabilize external factors for the comparison between the different algorithms, the distance from the head to the projection plane was fixed at 65 cm using a chin rest. The height of the chin rest was adjusted so that the eyes of the user were on a level with the upper edge of the virtual calibration grid (see Figure 5.9).

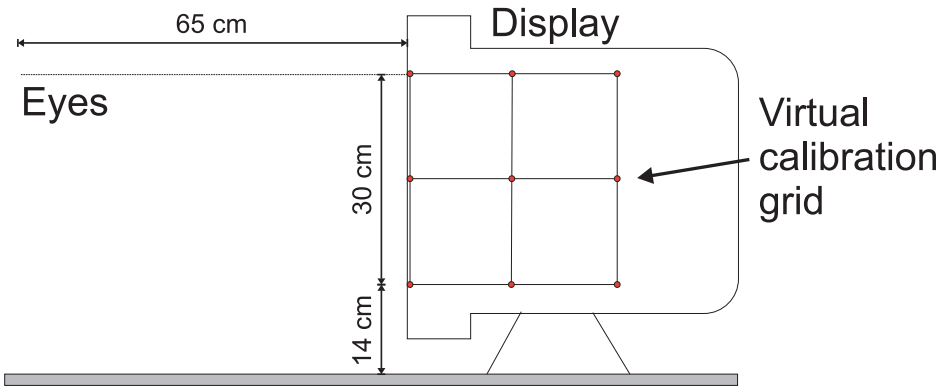


Figure 5.9: *Sketch of the set-up for the study (side view): the participant (left) gazes straight at the upper edge of the screen (right) from a distance of 65 cm. The virtual space fits exactly inside a cube with an edge length of 30 cm located behind the plane of projection.*

The two eye trackers, the SMI EyeLink I and the Arrington PC60, are both head-mounted. In addition to the eye tracker, the participants also had to wear the shutter-glasses. The combination of a projection technology requiring special glasses and vision-based eye tracking systems is delicate, as the cameras of the eye tracking systems cannot see clearly through the glasses. In the study, the cameras were positioned below the glasses with a free, but very steep perspective onto the eye. For the SMI EyeLink I, a special mounting for the glasses was constructed, as the original one interfered with the bulky head-mounted eye tracking system. The construction also allowed for an increased gap between the eyes and the glasses, so that orienting the cameras of the eye tracking systems was easier.

After the standard 2D calibration procedure provided by the accompanying eye tracking software, a 3D calibration procedure was run. For this a sequence of points from a 3D calibration grid was presented to the participants; for a side view see Figure 5.9. To fixate the leftmost calibration point on the front side of the cube the right eye of the user had to rotate 49.27° to the left, whereas the rightmost point was 32.19° to the right. To fixate all points on the back side of the cube, the right eye had to rotate 36.16° to the left and 22.15° to the right. To fixate a point in the upper center of the front side, the eyes had to converge 8.99°, and for a corresponding point on the back side 6.16°.

A pilot study had shown that each person needed an individual time span to acquire 3D perception with the projection technology used, so the calibration was self-paced. During the calibration procedure, all points of the grid were

Table 5.2: *Object dimensions (in mm) of the target set of objects used for the fixation and selection task. The numbers refer to the objects as specified in Figure 5.8.*

Object Number	x	y	z
1, 2	23	8	23
3, 4, 17, 19, 22	20	24	17
5	30	30	30
6	30	10	30
7, 18, 20, 21	20	24	20
8	20	34	20
9	20	60	17
10	20	20	34
11	20	17	24
12, 15	20	20	24
13, 14, 16	20	17	24

presented dimly lit and only the point to be fixated was highlighted. The points were traversed on a per plane basis, as recommended by Essig and colleagues (Essig et al., 2006). However, Essig and colleagues displayed the points one plane at a time, while in this study all points were shown simultaneously, but dimly lit, to improve orientation.

A life-sized virtual reality model of a Baufix structure was shown during the experiment (see Figure 5.8). The experimenter verbally referenced objects within the model which should then be fixated by the participants. As soon as they fixated the object, the participants affirmed this by pressing a key. The 3D fixation points were calculated internally for each fixation using both algorithms, and the results were logged. This was performed with each participant using the 22 objects depicted in Figure 5.9.

5.9 Study 2: Results

Overall, 10 participants (4 women and 6 men) were tested. Their mean age was 26.2 years; the youngest participant was 21 years and the oldest 41 years old. All participants had normal or corrected sight (contact lenses) during the experiment. They rated the difficulty of the experiment with 2.2 on a scale from 1 (very easy) to 6 (extremely hard).

Table 5.3: *Results comparing the different conditions. A significant difference of the means of the fixation depths was found in favor of the PSOM-algorithm.*

Results	Arrington		SMI	
	geom.	PSOM	geom.	PSOM
normally distributed	no $p < 0.001$	yes $p = 0.943$	no $p = 0.038$	yes $p = 0.661$
mean	-195.77 mm	-18.75 mm	-248.55 mm	-70.57 mm
difference btw. alg.	sig. $p < 0.001$		sig. $p < 0.001$	
nominal error	sig. $p < 0.001$	sig. $p = 0.005$	sig. $p < 0.001$	sig. $p < 0.001$
standard deviation	526.69 mm	96.92 mm	149.3 mm	60.06 mm

Four participants reported difficulties in fixating the virtual calibration crosses: they experienced problems getting the crosses to overlap for experiencing the 3D impression. However, a post-hoc analysis of calibration data and fixations revealed no significant differences compared to other participants.

Precision and Accuracy

The relative deviations of the calculated fixations from the real object positions (defined by the center of the object geometries) over all participants are shown in the bagplots (Rousseeuw et al., 1999) for the axes y and z (depth) in Figure 5.10.

The Kolmogorow-Smirnow test (Conover, 1971) showed that both datasets are not normally distributed. Therefore the Mann-Whitney-Wilcoxon test (Hollander & Wolfe, 1973) was applied to examine whether the absolute means of both datasets were significantly different and whether they differed significantly from the nominal values. An alpha level of 0.05 was considered significant (see Table 5.3) in all tests.

In the test series for the two eye trackers, the results for the z axis show that the means of the fixations approximated by the PSOM are significantly closer to the nominal value than those calculated by the geometric approach (5.10 from left to right). Still, all means differ significantly from the nominal value. The means of the results for the device from Arrington Research were closer to the nominal value than those from the SMI eye tracker (5.10 from top

to bottom). Thus it can be said that the device from Arrington Research showed a higher accuracy in our study.

The SMI device, however, achieved a higher precision, which is expressed in the lower standard deviations when compared to the device from Arrington Research. The precision using the PSOM algorithm is higher than the precision of the geometric algorithm for both devices.

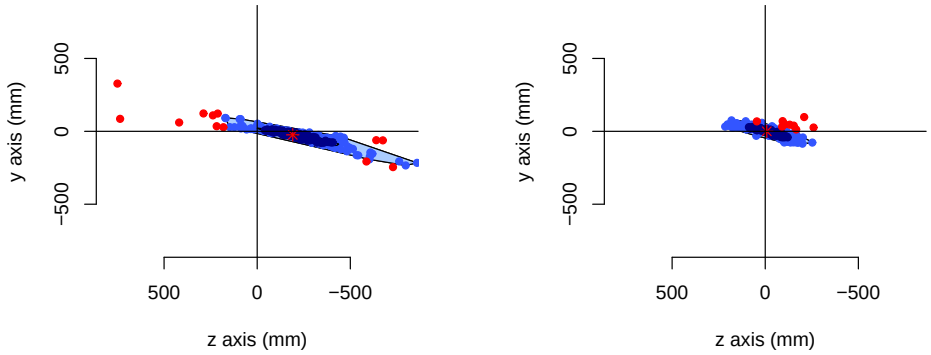
Performance on Pointing Dereference

Besides the described quantitative accuracy study, qualitative implications for applications were tested on a gaze pointing task. It was tested whether a dereferencing algorithm based on the 3D fixations manages to successfully identify more referent objects than an approach based on the direction of gaze only. Backed by the previous results, only the PSOM approach using the Arrington Research PC60 was evaluated in the gaze pointing task.

The direction-based dereferencing model determines the Euclidean distance between the 2D coordinates on the projection plane provided by the eye tracking software and the projected screen coordinates of the 22 objects (center of object). The object with the smallest distance to at least one of the fixations of both eyes was taken as the referent object. This is equivalent to the angular error measurement used in the manual pointing study, as the image shown on the projection plane is rendered for the perspective of the user; the distances measured are thus visually perceived distances. The referent object was then checked against the prompted object.

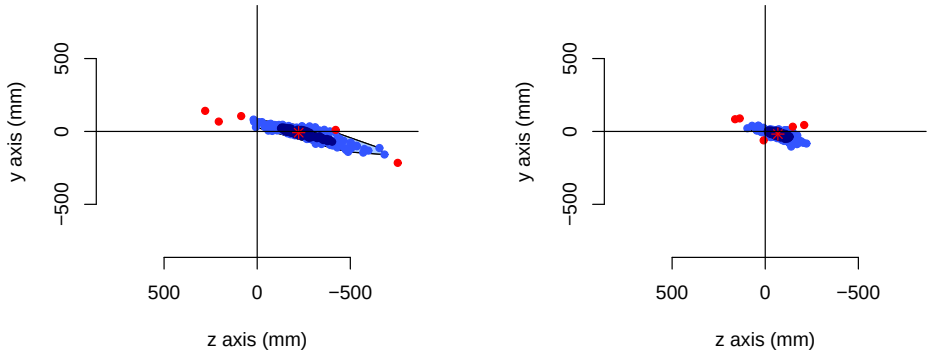
The location-based dereferencing model worked similarly using a standard 3D distance metric. Of the 22 objects, 4 were positioned in such a way that their projections partially occluded each other and thus led to an ambiguous situation for the direction-based model. This set of objects defined the critical area for the test.

The direction-based model successfully identified 165 (75%) of the 220 possible referent objects (22 per participant). The location-based model identified 92 (42%) objects. In the critical area with occluded objects (numbers 17 to 20), the location-based model managed to disambiguate 17 (42%) object selections, and the direction-based model only successfully identified 12 (30%) objects. Figure 5.11 shows the successful identifications per referent object. The numbering of the objects is depicted in Figure 5.8.



(a) Arrington PC60, Geometric Algorithm

(b) Arrington PC60, PSOM Algorithm



(c) SMI EyeLink I, Geometric Algorithm

(d) SMI EyeLink I, PSOM Algorithm

Figure 5.10: *Bagplots showing the relative errors of the different conditions for the y axis and the z axis. The perspective is equal to Figure 5.9, thus the user is looking from left to right towards negative z. The darker areas contain the best* 50% *fixations (those with the lowest deviations) and the brighter areas contain the best* 75% *fixations. The red dots mark outliers and the asterisks within the darker area marks the mean value. The extensions of the plots in z are smaller when the PSOM algorithm is used to determine the depth of the fixation (higher precision) and the mean values of the plots are closer to the center of the plot (higher accuracy). When the PSOM is used, the SMI system is more precise, but the Arrington system is more accurate.*

5.10 Study 2: Discussion

The following conclusions for the three hypotheses can be derived from the results of the study:

A: accepted. PSOM is more precise than the geometric approach The fixations approximated by the PSOM are significantly more precise and accurate than the results of the geometric approach for the y and z coordinates, for both eye trackers.

This result replicates the findings of Essig et al. (2006). Compared to their results, greater deviations of the means and of the standard errors were found. This was expected, as in the setting used in this study, objects were considered at distances between 65 cm and 95 cm from the observer, whereas Essig and colleagues used objects located in an area between 39 cm and 61 cm in front of the observer. They had already shown that the error increases with distance from the observer.

Further, in this study models of small real objects (diameter: $1^\circ - 3^\circ$ of visual angle), such as bolts and nuts, were used as referent objects instead of dots or crosses (diameter: 1° of visual angle). Thus the error, which is defined as the deviation of the fixation from the center of the object, will have a higher standard deviation because the participant can fixate on a larger area to refer to an object than when referring to dots.

B: rejected. The ViewPoint PC60 proved more accurate in the study Although the EyeLink I has a higher precision, the PC60 proved to be more accurate in this setting. One possible explanation could be that the 2D calibration using the shutter-glasses is more difficult with the EyeLink I because the adjustment of the cameras for the EyeLink I system is more difficult. In the study, the calibration fixations were often only rated as *poor* by the provided software. Thus the base data was less precise. This predication therefore holds only for the combination of eye tracking device, projection technology and shutter-glasses. However, while the results may not be transferable to different set-ups, they are still relevant for many desktop-based virtual reality set-ups that can be found in basic research. As a result, one cannot overemphasize that a good 2D calibration of the eye tracking system is critical for gaze-based interaction, and that special gear has to be developed that hosts stable perceivable motion tracking markers, glasses for the projection technology and eye tracking cameras in a comfortable way.

C: accepted. Considering fixation depth improves disambiguation for occluded objects The location-based model to dereference gaze pointing improved the disambiguation of occluded objects (objects 17 to 20 in Figure 5.8) from 30% to 42%. However, the performance of the location-based model on the whole scene yielded a lower success rate than the direction-based approach. This can be explained as follows: First, a comparison of the coordinates estimated by the location-based model with the coordinates provided by the eye tracker shows that the values estimated by the PSOM for x and y are less precise than those generated by the direction-based model. The direction-based model always chooses the best matching gaze direction, either from the left or the right eye, to trigger the identification. This should in most cases be the fixation from the dominant eye. If one gaze direction is noisy, the direction-based model is not affected. The PSOM depends on both gaze directions and thus is affected by inaccuracies of the subdominant eye. Second, the self-paced 3D calibration was not coupled with a quality control as the 2D calibration is. This could have led to less than optimal calibrations for the PSOM approach. The accuracy of the PSOM approach could thus be further increased by improving the detection of the dominant eye and by optimizing the calibration procedure.

The results show that 3D fixations can be derived from vergence movements, and the findings of Essig et al. (2006) can be generalized to more realistic virtual reality scenarios, as the present setting with the objects from the toy building block set demonstrates.

The adaptive PSOM approach based on five parameters (x/y coordinates of left and right 2D fixations and difference in x coordinate) outperforms geometric triangulation. However, the performance shown in this study is not as good as in the study of Essig et al. (2006). This can be attributed to larger fixation target sizes and the more difficult set-up: in the original setting, the fixation targets were distributed over four levels of depth, two behind the screen (-3.67 cm and -11 cm) and two in front of the screen (3.67 cm and 11 cm). The fixation targets in this setting were all presented behind the screen (-4.5 cm to -25.5 cm). Fixations closer to the user require greater vergence movements and thus measurement errors have a smaller effect.

External factors, for example the virtual reality technology used for the study, may limit the performance. Insufficient channel separation (ghosting) of the applied stereoscopy method and a tracking from below the glasses complicates the procedure. More advanced technologies, such as passive projections based on polarized light, could thus further improve the performance. Also, the limited interaction space of the desktop-based VR platform led to a crowded

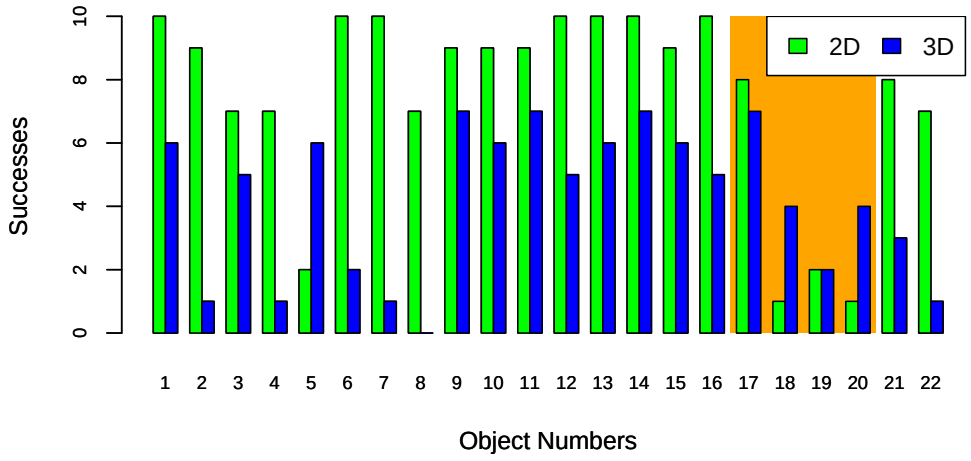


Figure 5.11: *Histogram of the correct referent identifications over all 10 sessions (see Figure 5.8). The critical area of overlapping objects (numbers 17 to 20) is highlighted. While the location-based pointing model is outperformed by the direction-based model in unambiguous cases, it achieves better results in the ambiguous cases, especially for objects 18 and 20.*

scene, reaching the limits of the resolution of the eye tracker. In this study, 22 objects were used as fixation targets on a single display. Most studies in basic research that employ eye tracking on a computer screen, restrict themselves to four to eight objects on one screen.

Location-based dereferencing of gaze pointing has shown first successes when occluded objects need to be disambiguated. Several weaknesses, such as the dependency on the subdominant eye and the need for a quality control for calibration, have been identified that define concrete starting points for further optimizations. For current applications, a hybrid approach that uses direction-based gaze pointing per default and disambiguates occluded objects using location-based pointing in cases of ambiguity could be a viable solution.

5.11 Visualizing the Point of Regard in 3D

Eye gaze movements on 2D surfaces, such as web pages, pictures or videos have been analyzed in basic research and usability studies for years. Chapter 2

has presented two different visualization techniques for the distribution of attention on 2D surfaces (see Figure 2.11 and Figure 2.12). Yet, the increased interest in investigating real-world interactions demands for more flexibility in the scenery to be analyzed. Not all scenarios can be mapped to a 2D computer screen and the manual analysis of videos taken by a scene camera during online interactions is time-consuming and error-prone.

Applications for the technologies brought together in the interaction framework and tested in the studies are thus not restricted to human-computer interactions. There is indicated interest in basic research, for example in visual attention, linguistics or sports, as well as in the industry, for example to test product designs or optimize the point of sale. These disciplines require descriptive visualizations of the gathered data. The following visualization has been developed for the description of 3D points of regard in analogy to the established scanpath technique for attention on 2D surfaces.

5.11.1 3D Scanpaths

Scanpaths of individuals are depicted as a sequence of dots marking the target of fixations, which are overlaid on top of the image. A similar procedure can be applied to the visualization of the point of regard in 3D as well. To give an impression, Figure 5.12 shows an ideal scanpath for the gaze pointing task used in the study on location-based pointing. This visualization is not based on gaze data, but depicts the locations of the target referent objects as well as their sequence during the task.

The 3D model of the Baufix assembly that was used in the study is defined in X3D, an ISO-standard (ISO 19775-1:2004, 2004) for describing 3D worlds in a scenegraph. This description is the starting point for the visualization. The points of regard and links in between them are added automatically by the developed visualization algorithm based on the eye tracking data. For this, the algorithm instantiates external X3D prototypes whose definitions can be altered to change the appearance of the visualization.

Figure 5.13 provides an overview of the results of the different algorithms and eye tracking devices for participant 4. What is visible in the 3D scanpaths but could not be seen in the statistics presented so far: while the errors of the geometric triangulation are greater than the errors of the PSOM approach, they are systematic, and the structure of the underlying assembly of Baufix parts can be recognized, albeit spatially distorted in depth. This is an added

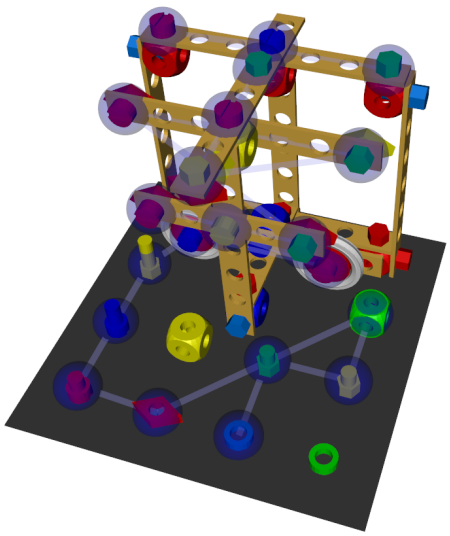
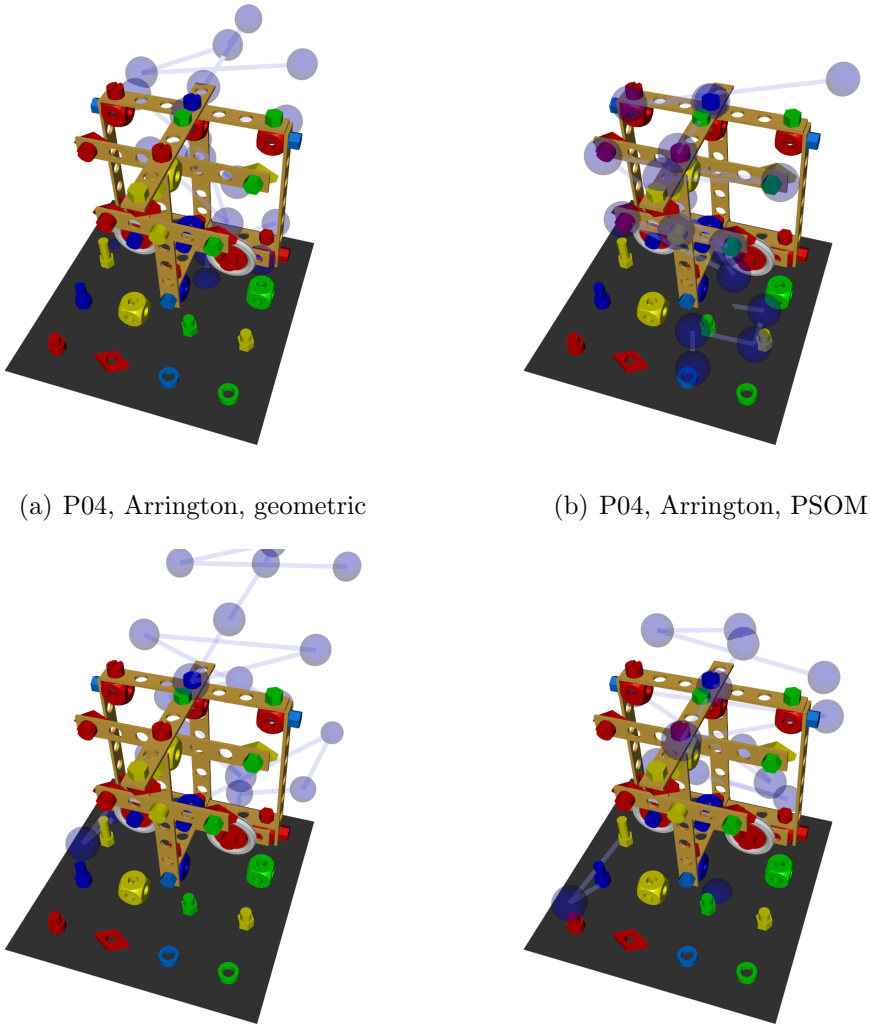


Figure 5.12: *The picture shows a 3D scanpath on the Baufix assembly used for the study presented in Section 5.6. The scanpath depicted is the purely hypothetical optimal scanpath for the target objects. Not all objects were used in the instructions, thus some, such as the green ring at the lower right corner and the yellow block in the center, are excluded from the scanpath.*

value the 3D scanpath visualizations provide to the scientific evaluation of the results.

3D scanpaths can also be used to visualize the data aggregated over all participants of a study. The results are shown in Figure 5.14. These aggregated scanpaths provide a feedback of the overall distribution of the points of regard, but they lack the clarity of the individual scanpath visualizations. It is neither possible to identify individual scanpaths, nor can the amount of points of regard on a specific object be estimated well enough to provide further insights. Thus, 3D scanpaths naturally face the same problems as 2D scanpaths. The attention volumes presented in Section 6.4.1 overcome these problems.



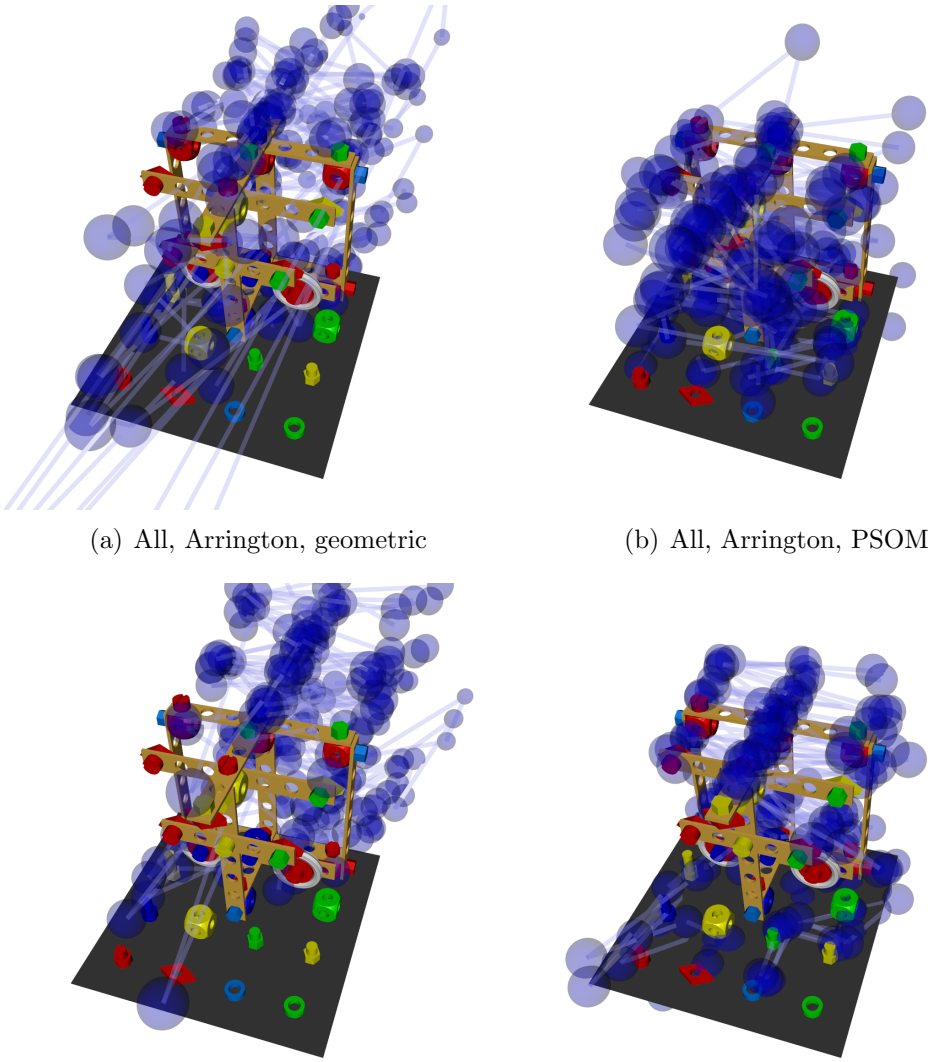
(a) P04, Arrington, geometric

(b) P04, Arrington, PSOM

(c) P04, SMI, geometric

(d) P04, SMI, PSOM

Figure 5.13: *A comparison of the 3D scanpaths of participant 4. The model is rotated in these visualizations to provide a better viewing perspective. The participants gazed at the model from the lower left from a distance twice as long as the width of the base plate. In the visualizations, the difference between the geometric triangulation and the PSOM approach can be clearly seen. The results from triangulation are spread more in depth than the PSOM results. Yet the structure of the underlying Baufix assembly can be more or less recognized in all visualizations.*



(a) All, Arrington, geometric

(b) All, Arrington, PSOM

(c) All, SMI, geometric

(d) All, SMI, PSOM

Figure 5.14: *If more than one scanpath is depicted, the clarity of the picture is reduced. While the extension of the area where points of regard have been detected can be determined, the areas that get the most attention cannot be retrieved and the objects themselves get obscured.*

5.12 Summary

The studies presented in this chapter attest that gaze-based pointing can be used for human-computer interaction in immersive virtual reality. Additionally, it was shown that the prototype of the interaction framework for gaze-based interactions DRIVE (see Chapter A) provides reasonable latency and accuracy for HCI. This was demonstrated on the *Visual Ping* test, which was devised to estimate the latency of the system during a gaze pointing task. The total latency of the system was about 300 ms, and the approximated latency of the DRIVE framework about 70 ms.

For testing *direction-based* gaze pointing, a pointing cone model for the extension of pointing with an aperture angle of 10° was used. The mean horizontal accuracy of the gaze direction was 1.18° and the mean vertical accuracy 2.52°, both very accurate measures within the expected ranges. Summing up the findings for direction-based gaze pointing, the DRIVE framework in combination with the deployed tracking technology supports an accurate detection of gaze pointing and offers a high success rate for identifying the referent objects (see Figure 5.7).

Whereas manual pointing gestures can only be dereferenced using direction-based approaches, gaze pointing also allows for *location-based* models if binocular eye trackers are available. The additional information about the depth of the point of regard is valuable, for example to disambiguate between overlapping possible referents. The naïve approach using geometric triangulation does not provide sufficient accuracy. The more advanced PSOM approach tested in study 2 (Section 5.6) provides an improved accuracy. The point of regard can be located in 3D up to a mean error of M=1.88 cm (SD=9.69 cm). Location-based pointing, however, requires a longer 3D calibration procedure to provide the parameterization of the PSOM. The 40% increase in discriminative power for the disambiguation of occluded objects which was achieved by the location-based approach shows room for further improvements, and promising approaches have been discussed.

The detection of the point of regard in 3D space provides relevant information which can be of great benefit for other research disciplines and commercial applications beyond gaze pointing in conversational interfaces. The 3D visualization presented in the preceding section provides valuable feedback to the researcher. Together with the motion capturing and the interaction framework for recording, DRIVE (see Chapter A), tracking visual attention is now detached from flat surfaces and can be applied to real objects as well (see also Section 7.1.4).

Chapter 6

Modeling the Extension of Gaze and Manual Pointing

This chapter presents precise models for *gaze and manual pointing*. Section 6.1 reconsiders the corpus on manual pointing gestures which was collected in the study presented in Chapter 4. It provides essential reflections on the way the corpus was initially analyzed and presents an alternative frame of reference. The proposed frame of reference breaks with the segmentation of the pointing domain into discrete rows and introduces a continuous measure based on the distance from the finger tip to the referent. This change in the basis of the analysis lays grounds for the development of generalized models of manual pointing.

Answering the *where*-question for manual pointing The question on *where* interlocutors are pointing to is approached in Section 6.2, which is concerned with finding a model to describe the pointing direction. An accurate model of the direction of pointing is a fundamental requirement for models of the extension of pointing. To these ends, this section presents data on a comparison between the Index-Finger Pointing model (IFP) and the Gaze-Finger Pointing (GFP) model (see Section 3.2.1), based on the corpus on manual pointing.

The basis for this comparison are measurements of the *precision* and *accuracy* in predicting the ideal direction of pointing which are achieved by IFP and GFP. A first comparison of IFP and GFP presented in Section 4.9.2 already came to the conclusion that the GFP model is a better approximation of the ideal pointing direction. The results, however, were not optimal, indicating

that the validity of the GFP model is not optimal. This comparison is reconsidered based on the new frame of reference.

As a consequence of the previous results, Section 6.2 also introduces more sophisticated variations of the GFP model which include information about eye dominance. The evaluations confirm that the refined model, called GFP/dom, produces accurate predictions of the direction of pointing and thus constitutes an adequate candidate for modeling the spatial extension of manual pointing.

Answering the *which*-question for manual pointing Section 6.3 starts with a series of tests of models for the spatial extension of pointing, such as the vector extrapolation model (see Section 3.3.2) and the shape-based pointing cone model (see Section 3.1.4.2 and Section 3.3.3), on the corpus of manual pointing. Both models are parameterized with the GFP/dom model for the optimal pointing direction. Based on these findings and the observations of the dichotomization of the pointing domain into a proximal and a distal area, a hybrid model is developed that combines the advantages of the pointing cone model and the distance-based approach.

Answering the *which*-question for gaze pointing Section 6.4 turns towards pointing models for gaze pointing. The two studies on direction-based and location-based gaze pointing presented in Chapter 5 have already testified that gaze pointing is accurate and precise, especially when compared to manual pointing. The section thus reconsiders the results from the study on location-based gaze pointing, provides a refined model for gaze pointing and finally presents *Attention Volumes* as a new visualization method for the distribution of attention in 3D.

Integration into a conversational interface In addition to the empirical findings on human pointing which stand on their own, the aim of this thesis is to use the models of pointing to improve conversational interfaces and make interactions with the machine more natural. Section 6.5 describes how the developed models of pointing can be used in one specific conversational interface. This section provides an example of an integration of the models in a constraint-based satisfaction approach to resolve multimodal deictic expressions. This example also concludes the contributions of this chapter to the thesis.

6.1 Study on Manual Pointing Reconsidered

One drawback of the analysis of the study on manual pointing presented in Chapter 4 is that it uses the underlying grid layout of the pointing domain as a distance measure. Among other reasons, this decision was motivated by the fact that the presented study started as a replication of the previous study of the DEICON project in the CRC 360, and thus the same frame of reference was used for the analysis. This decision, however, makes it difficult to transfer the findings to other domains. The discrete grid-scale generalizes too much over the recorded data, as referent objects in the same row can be targets of pointing acts of quite different morphologies.

The visualizations of the Gesture Space Volumes presented at the end of Chapter 4 illustrate that the participants of the study exhibited different behaviors with respect to *co-verbal* manual pointing and *uni-modal* manual pointing. About 61% participants used the *leaning-forward strategy* to reduce the distance between the tip of their pointing finger and the referent. These new insights about the coping strategies motivate a break with the discrete grid-scale. Hence, the following analysis steps out of the self-imposed grid-constraints by using the distance between the finger tip and the referent as frame of reference.

6.1.1 Distance between Finger Tip and Referent

The Description Givers (DGs) adjusted their upper body, e.g. by leaning forward, to extend their gesture space further over the domain of possible referents. Thereby they dynamically changed the distances between finger tip and referent on a per move basis. The distance is individually different for two DGs pointing to the same referent object. During the S+G trials, the pointing hand of the DG remained closer to the upper body in most cases. In the G trials, its extension was generally increased. To take these variances into account, each pointing move is considered separately – neither aggregated over rows nor over objects – and distances are computed accordingly.

The qualitative impression of the effects of the different pointing strategies on the morphology of the 3D gesture space has already been shown in Figure 4.17 (page 94). The results of quantitative evaluations are depicted in Figure 6.1. The plot shows how close the DGs moved their finger tips to the referent as a function of the target row. Overall, when pointing at referents in the distal area during the G trials, the DGs extended their finger tips in the

mean about 0.30 cm further than in the S+G trials. In the S+G trials, the DGs moved their finger tips close to the referent up to row 3, which was at a distance of 47.75 cm from the edge of the table. This is roughly the length of a human forearm and hand. In the G trials, the DGs invested more effort and maintained a short distance to the referent until row 4, which was at 67.75 cm. In normal seating position with the shoulders about 10 cm behind the table, the arm had to be extended by about 85 cm to nearly touch referents in row 4 (without leaning forward). It is thus reasonable that leaning forward started from row 4 on with individual differences, e.g. based on the personal height when sitting. In row 8 the mean distance between finger tip and referent exceeds one meter. Row 8 is 147.75 cm from the front edge of the table or approximately 157.75 cm from the shoulder of the DG. The boxplot for the G trials in row 8 shows that about half of the DGs reduced the distance even further, which they could not have managed without leaning forward.

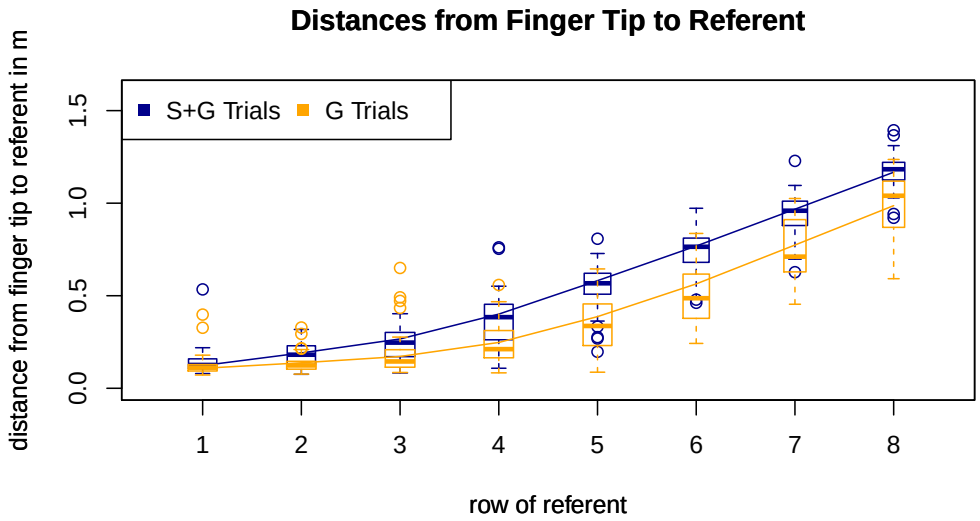


Figure 6.1: *The leaning-forward behavior found as a coping strategy in the G trials of the manual pointing study (see Figure 4.17) can be identified in the plot shown above. It depicts the distances from the tip of the DG's pointing index finger to the referent as a function of the row of the referent.*

The change in the frame of reference clearly affected the distribution of the number of identifications, which are now accumulated per distance. In the old grid-scheme, 4 objects needed to be identified per row, which amounts to 100 identifications per row over all 25 participants. For the distance-based frame of reference, the distribution is more heterogeneous. Figure 6.2 shows the number of identifications as a function of the distances the DGs covered with their finger tips during the stroke in the S+G and G trials. As can be

seen, for the majority of the identifications, the finger tip was close to the referent. This was even more so for the G trials. But the graphs also show something else: they highlight the identification failures of the OI. Failures occured only in the G trials, and only if the tip of the index finger was at least 30 cm from the referent. Based on the data, it can be assumed that manual pointing in this scenario started to get ambiguous at 30 cm from the finger tips. This, however, is surely a domain-specific effect which depends on the distribution of the possible referents. As such it has to be considered with care and cannot be generalized. Nevertheless, the DGs quite successfully tried to reduce the distance below this critical threshold as if they were aware of it, and they automatically adapted their pointing behavior to the situation.

In the scenario used for the manual pointing study, pointing gestures are unambiguous if the index finger is less than 30 cm away from the referent.

A second strategy which was used by 48% of the DGs is *raising high*. The distribution of the number of identifications relative to the height of the index finger during the stroke is depicted in Figure 6.3. During co-verbal manual pointing, the height levels between 2 cm and 30 cm are represented equally well, there is no clear preference. This changed in the G trials, where the height levels below 10 cm were clearly preferred. Above 10 cm the OIs started to fail with their identifications. In addition, it can be observed that during the S+G trials the maximum index finger position was below 38 cm, while in the G trials the height levels extended up to 48 cm. Maintaining a higher position, however, does not seem to be a very good strategy, as all failed identifications concern heights above 10 cm. This is detailed further in the scatterplot of the heights as a function of the distance shown in Figure 6.4. The failed identifications are brushed in red. The failures are clustered in an area starting at 10 cm above the table and 40 cm away from the DG.

The *leaning-forward* strategy is successful in coping with the short range of distinctiveness of manual pointing in this scenario. The *raising-high* strategy leads to an increase in misinterpretations on the part of the OIs.

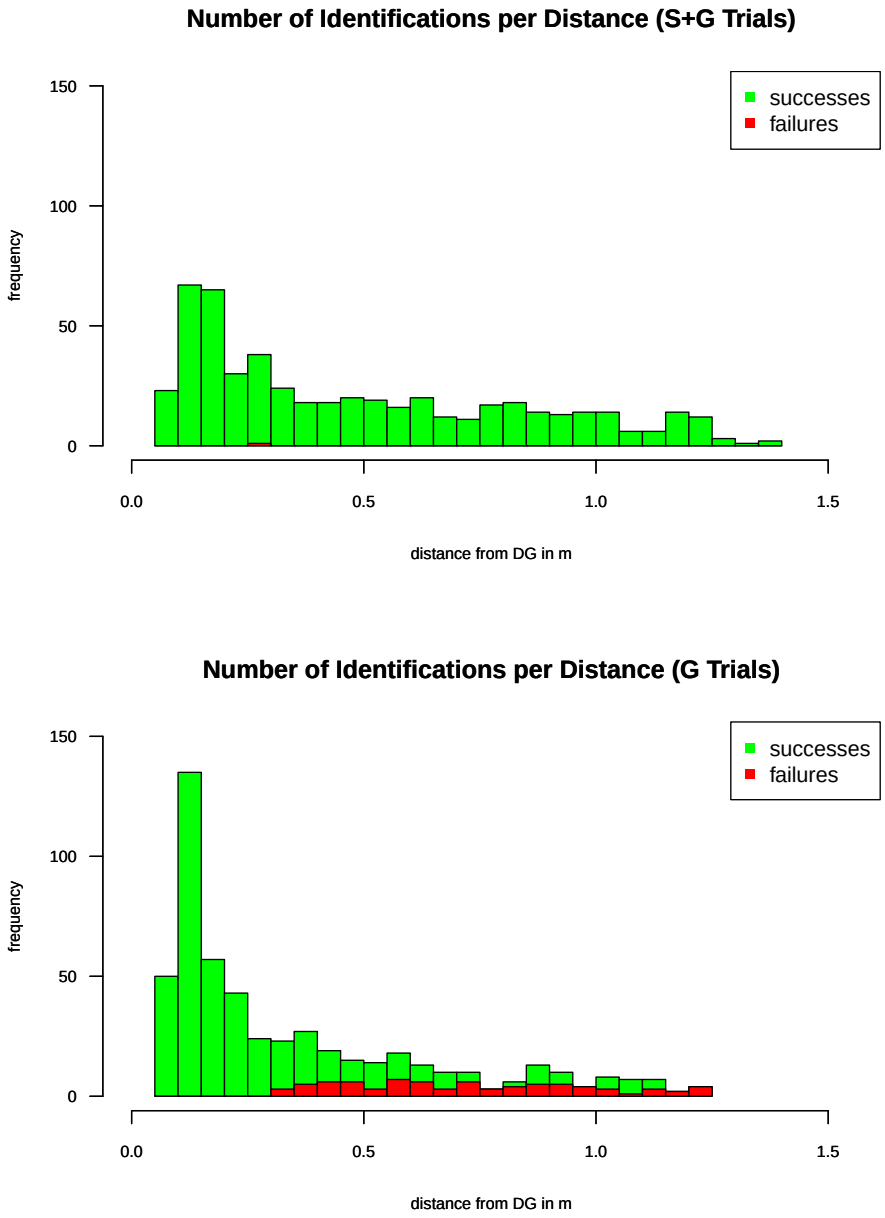


Figure 6.2: *In the G trials, the OI only failed to identify referents which were relatively far from the DG. This was already shown in the row-based analysis. The distance-based analysis reveals that failures increased with the distance from the finger tip, starting from a minimum distance of 30 cm. Pointing gestures to referent objects in row 5, for example, did not fail if the DG leaned forward to reduce the distance between finger tip and object below 30 cm.*

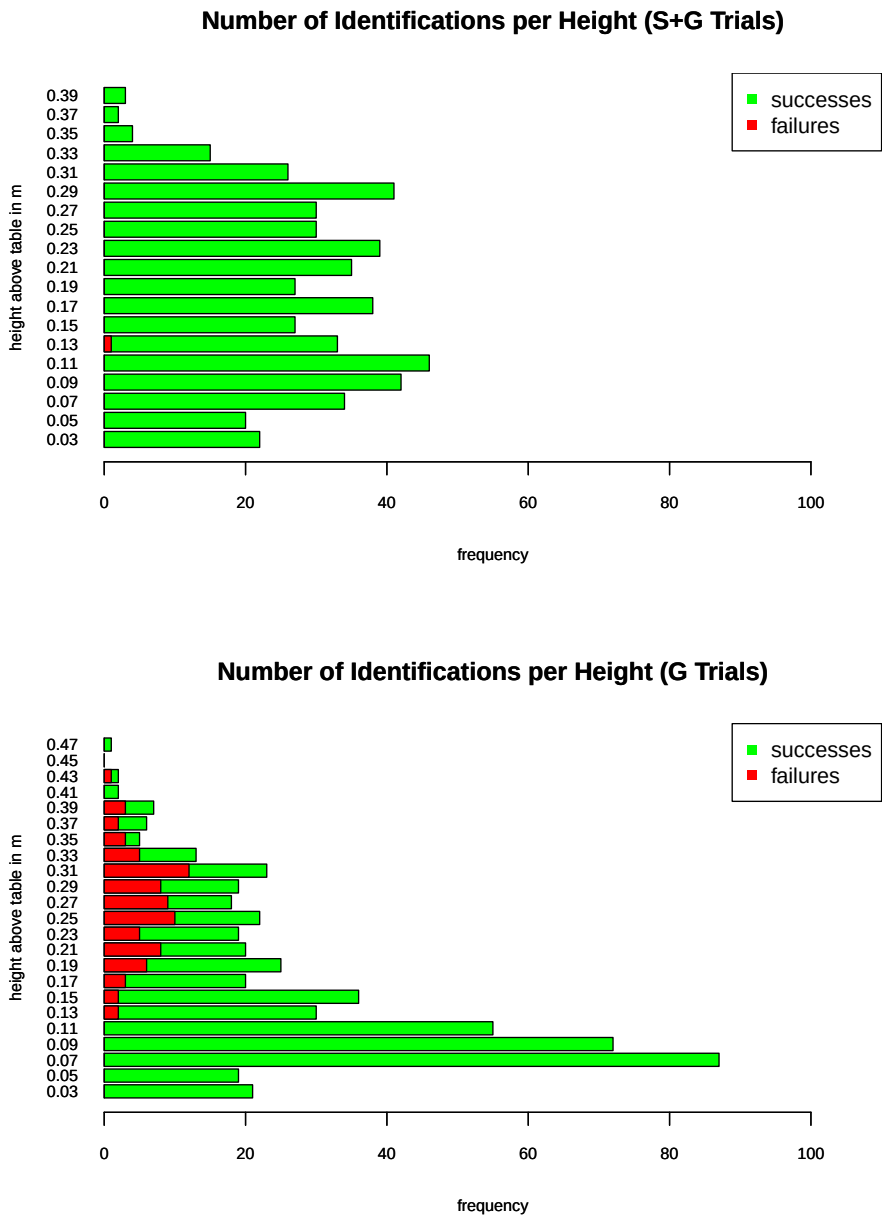


Figure 6.3: *During co-verbal manual pointing, the index finger is held at different heights above the table and – as can be seen in the plot at the top – the height levels between 3 cm and 33 cm are equally distributed. In the G trials, the index finger is held more often at a lower height level and this coincides with successful pointing acts. Above a height level of 11 cm failures increase in the G trials.*

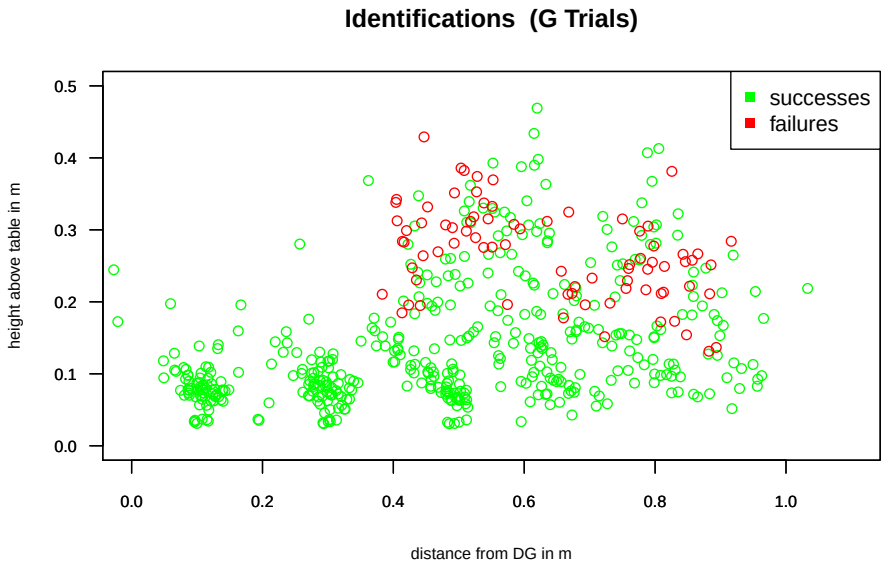


Figure 6.4: *The demonstration acts in which the OI failed to identify the referent object are distributed over the upper distal area of the 3D gesture space (top right area in the graph). The locations of the index fingers are depicted as circles, green circles mark successful and red circles mark unsuccessful pointing acts.*

6.2 Modeling the Direction of Manual Pointing

A precise model of the direction of pointing is the essential basis for models of the extension of pointing. The GFP model used so far estimates the direction of manual pointing based on a vector from a point between the eyes, the "cyclopean" eye, towards the tip of the index finger. This first approximation is efficient and converges on the ideal pointing direction better than the IFP model. Nevertheless, the obtained accuracy is still far from optimal. The imagined cyclopean eye used in the GFP approach is a simplification. The two human eyes do not contribute equally to aiming at the referent object. Humans have a preferred eye, the *dominant* eye, and sometimes they even shift dominance between eyes depending on the context. Banks, Ghose & Hillis (2004) for example attested a positive effect of perceived relative image size on eye dominance.

It thus seems advisable to further optimize the GFP model by taking eye dominance into account. For this, different submodels of GFP have been implemented and tested on the data from the manual pointing study. In the following, the original GFP model is called *GFP/cyc(lopean)*, the GFP variants for a preferred left or right eye are *GFP/left* and *GFP/right*. In addition, a dynamic GFP model *GFP/dom* is introduced which switches dominance between the eyes based on the context. A coherent model for eye dominance switching is not available at the moment, so the current GFP/dom model uses a post-hoc test based on the knowledge about the current referent to evaluate the accuracy of the GFP/left and GFP/right models. Based on the results of this test, the GFP/dom model switches eye dominance to the eye whose GFP model led to the best approximation of the pointing direction. The GFP/dom model thus demonstrates the performance that can theoretically be achieved by a GFP model with dynamic dominance switching if the dominant eye can be either detected or reliably predicted.

6.2.1 Analysis of Accuracy

Figure 6.5 shows the distributions of the angular errors for all models for the S+G trials. The closer the peak of the model curves are towards zero and the smaller the curve, the better the corresponding model approximates the direction of pointing. The IFP model, for example, has a rather broad maximum with a peak at about 14°. The original GFP, GFP/cyc, has a peak at about 4° and is also more narrow than IFP. The GFP/left model is worse than GFP/cyc, but the GFP/right model shows a very good performance. GFP/dom and GFP/right show a peak performance at 0°, which means that they are very good approximations of the ideal direction of pointing. As eye dominance in the right eye is more common (not only in this study, but also generally, see Chaurasia & Mathur, 1976), GFP/dom uses the GFP/right model for most pointing acts.

> The specialized GFP/dom model is a good approximation of the ideal direction of the pointing gesture. It clearly outperforms the previous models IFP and GFP in predicting the direction of manual pointing.

At the moment, however, a model to predict switches of eye dominance is missing and GFP/dom can only be used post-hoc once the target referent is known. As a good approximation, the GFP model for the preferred eye can be chosen (here GFP/right) for online use in human-computer interaction. A closer examination of the pointing acts where GFP/left performs better than

GFP/right reveals that 63% of these acts were performed left-handed (see Figure 6.6). The GFP/left model only provides better results in one single case of right-handed pointing gestures. So a change in the pointing hand might also indicate a switch of eye dominance.

If eye dominance cannot be measured or detected online, GFP/left or GFP/right can be used to approximate GFP/dom, depending on the default eye dominance. When pointing, also the laterality of the pointing hand can be used to indicate the dominant eye.

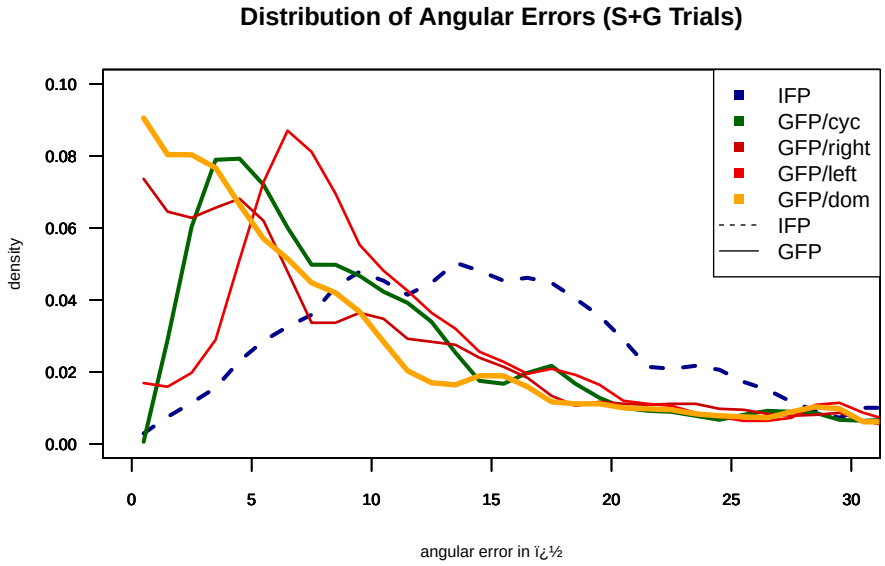


Figure 6.5: *The graph shows a comparison of the accuracy of the models for pointing direction. Shown is the density of angular errors. An optimal solution should have a peak at zero. The GFP/cyc model comes close to zero with a peak at* 7.5°. *The GFP/dom model has the best accuracy, closely followed by GFP/right, which outperforms GFP/left because the DGs were right-handed.*

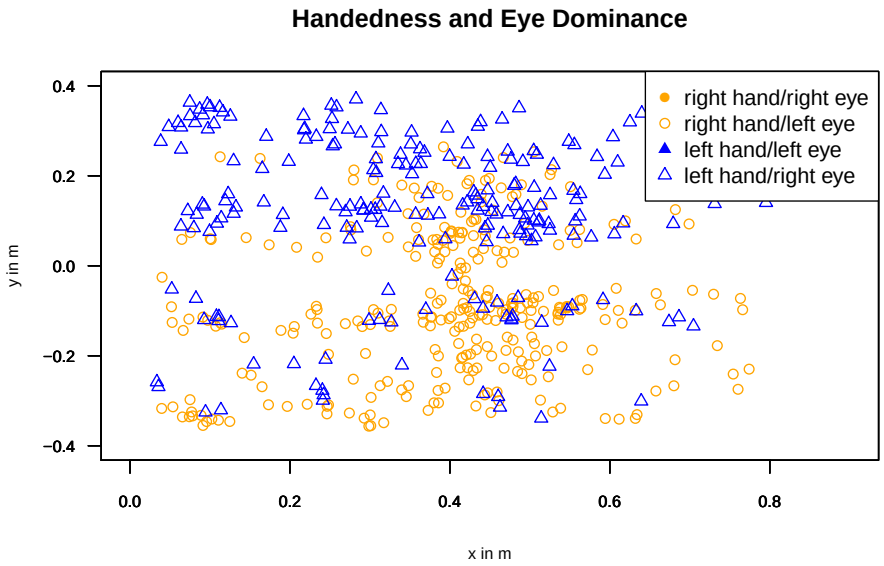


Figure 6.6: *This graph shows the distribution of right- and left-handed pointing gestures. For all pointing acts, the positions of the finger tip during the stroke are marked. If the direction of pointing is described best by the GFP model which is equilateral to the pointing hand, the symbol is solid, otherwise it is only outlined.*

6.2.2 Analysis of Distance Dependencies

The main motivation for reconsidering the manual pointing study was to tell which model for the direction of pointing, either IFP or GFP, is better. This could not clearly be answered in Section 4.9.2, which is *inter alia* reflected by the graphs depicted in Figure 4.13 (page 87), where the graphs for IFP and GFP in the G trials are not easily disentangled in the distal area.

The differences between IFP and GFP stand out much clearer than in the original graph in Figure 4.13 if the mean angular and orthogonal errors are plotted as a function of the distance between the finger tip and the referent. Figure 6.7 shows the data smoothed by the LOWESS smoother (Cleveland, 1981). In the S+G trials, the angular errors of IFP approximate 11.7° in the distal area, while GFP/cyc achieves lower angular errors of about 6.5°. The most accurate results are generated by the GFP/dom model with 4.1°. A similar qualitative behavior can be found in the G trials, but with lower quantitative differences (IFP 10°, GFP/cyc 8.7° and GFP/dom 7.3°).

The pointing direction can be predicted best by the GFP/dom model, with an error as small as 4.1° in the distal area.

The shapes of the curves for the angular errors suggest a distinction between *proximal and distal pointing*. Within the first 40 cm the curves show a steep negative slope with angular errors going down from 20° and 40° to about 10°. Beyond 40 cm the curves show a nearly asymptotical course. In the G trials, GFP/cyc and GFP/dom provide distinctly better results in the proximal area, but for distances above 40 cm, IFP and GFP models provide nearly identical results. As a consequence, it could be necessary to distinguish between proximal and distal pointing in the pointing models to do justice to the observed differences.

In the G trials, the DGs extended their pointing hands more often to get as close as possible to the object. This amounts to fewer pointing acts with larger pointing distances, as has been shown in Figure 6.2. By implication this means that for pointing distances greater than 40 cm, the arm of the DG is already extended, the line of gaze and the direction of the index finger are nearly conform and thus also the different error measurements coincide.

This is different for the S+G trials, where the DGs also point at distant objects with a nearly completely flexed elbow. In doing so, they do not reduce the distance between finger tip and referent. Instead, they reduce the distance between their dominant eye and the finger tip. This makes aiming with the dominant eye more accurate, but at the same time small errors have a larger effect and thus models with an incorrect or only approximated origin of the gaze will provide larger errors. This behavior is directly observable in the graphs for the S+G trials showing differences between the three models which are larger than those found in the G trials.

The closer the referent gets, the lower the accuracy of the GFP models.

In the S+G trials, differences between IFP and GFP models show up in both orthogonal and angular measurements. The orthogonal errors of IFP increase about twice as fast with increasing distance than those of GFP. The slope of the orthogonal errors measured for IFP is also greater for the S+G trials than for the G trials. Also, the angular errors of IFP are higher than those of GFP and even higher than the angular errors in the G trials. The GFP/dom model, on the other hand, produces the smallest angular errors for S+G trials, even smaller than for the G trials.

The GFP/dom model is the most accurate model in describing the direction of pointing over all trials.

The GFP/dom model constitutes a very accurate description of the direction of human manual pointing gestures. This model also does not require much additional effort in its application compared to the GFP/cyc model. The distance between the eyes of the user can be easily obtained – for the stereo projection in 3D environments this distance is already required – and knowledge about the defaults regarding the dominant eye of the user can be used as a good approximation, if no model to dynamically predict changes in eye dominance exists. As a consequence, the optimization of the model for the direction of pointing is considered as settled. The following section proceeds with the evaluation of models for the extension of pointing.

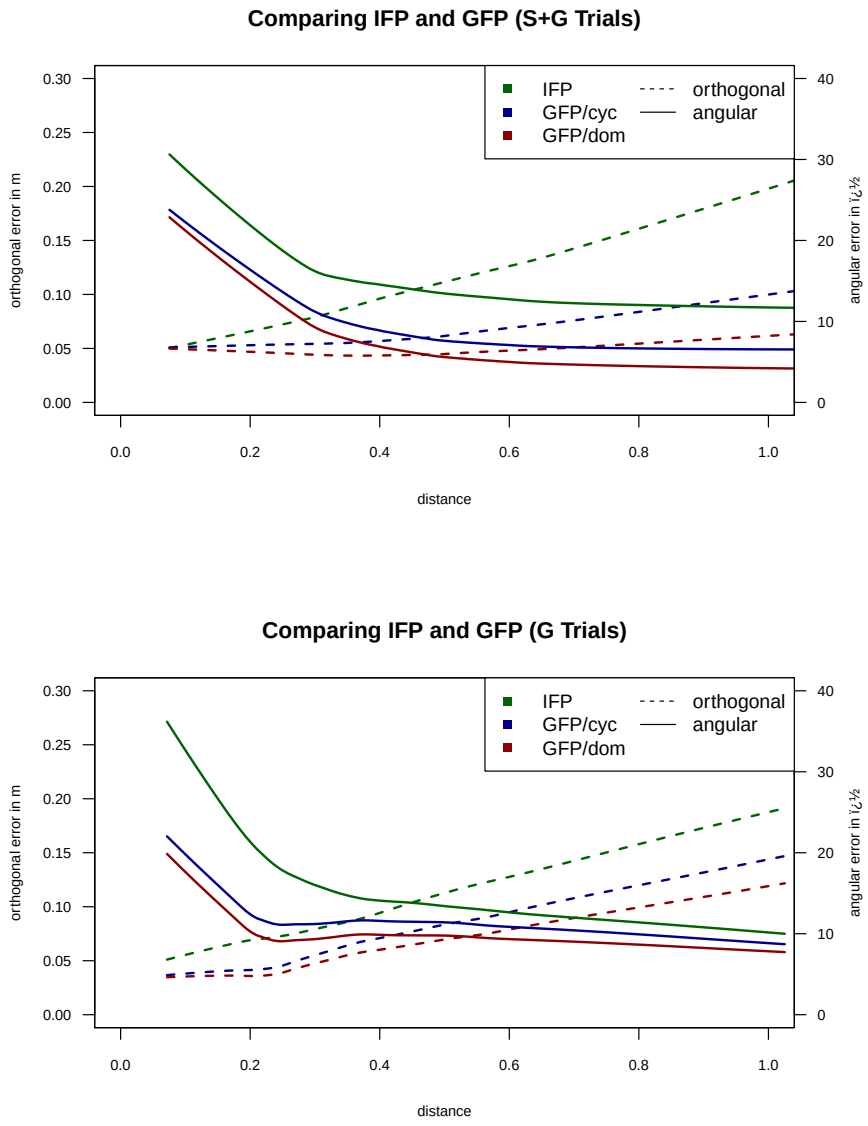


Figure 6.7: *The angular and orthogonal errors produced by the different models for the pointing direction (IFP, GFP/cyc and GFP/dom) are compared in these diagrams for the S+G trials and G trials. Overall, GFP/dom produces lower errors in all trials and for both measurements.*

6.3 Modeling the Spatial Extension of Manual Pointing

In the previous section, alternative models for the direction of pointing were developed which take into account the dominance of an individual's eyes. From these models, the GFP/dom model, which dynamically switches between the eyes, emerged as the best solution ($p = (\vec{o}_{tip}, \vec{v}_{GFP/dom})$). This optimized model for the direction of pointing can now be used to parameterize and evaluate the models of the spatial extension of manual pointing. In the following, the *Vector Extrapolation Model* and the *Pointing Cone Model* will be tested.

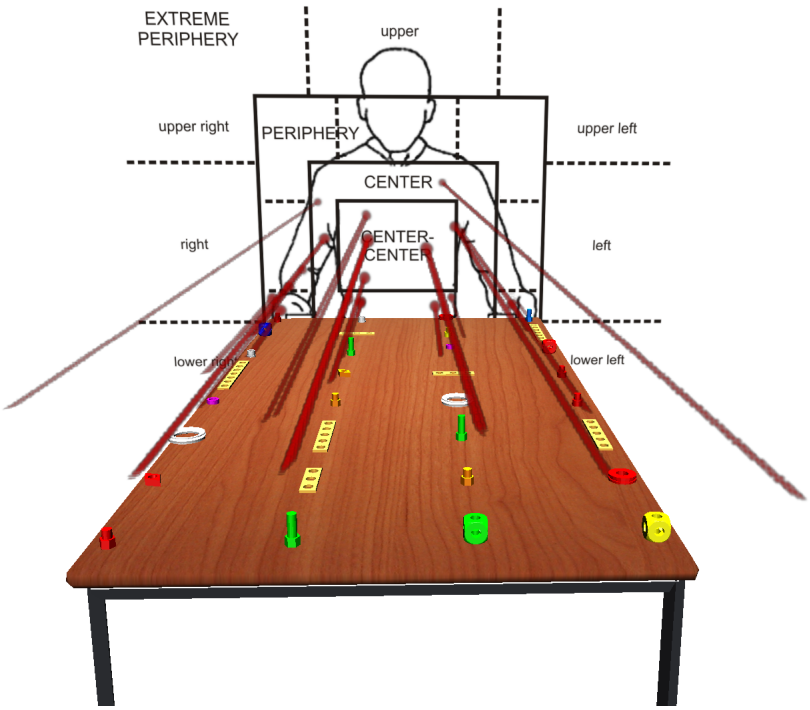
6.3.1 The Vector Extrapolation Model

Based on the GFP/dom model, the inital equation of the vector extrapolation model (see Equation 3.7 on page 54) can be refined to:

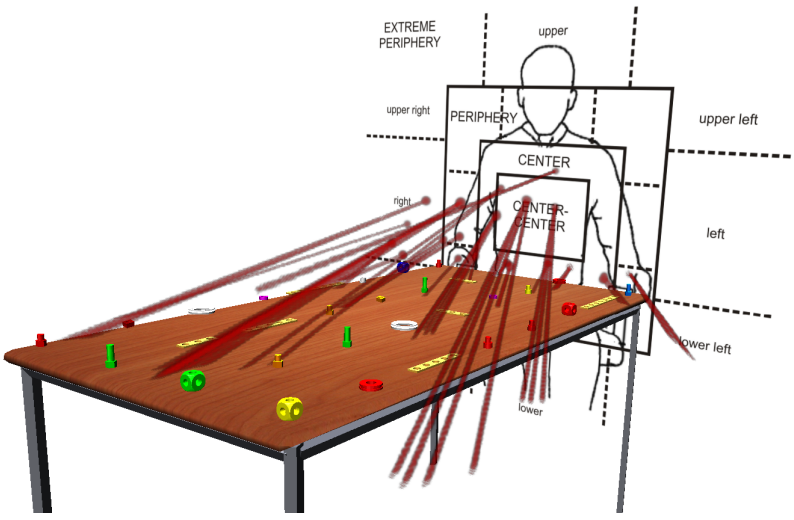
$$\begin{aligned} S_{vector}(p) &:= \{r|(G(r) \cap \vec{o}_{tip} + d\vec{v}_{GFP/dom}) \neq \emptyset\} \\ \text{with } \vec{v}_{GFP/dom} &:= \vec{o}_{tip} - \vec{o}_{eye/dom} \end{aligned} \tag{6.1}$$

The vector extrapolation model requires that the referent is hit by the vector defined in Equation 6.1. If the findings presented so far are considered, it is evident that vector extrapolation will not successfully dereference many pointing gestures. This can be seen in the distribution of the intersections of the pointing rays with the table presented in Figure 4.11 and the distribution of the angular errors depicted in Figure 6.5. Figure 6.8 shows the predictions of the vector extrapolation model applied to the data recorded for a single individual. Over all participants, the vector extrapolation model identified 9.7% of the referents in the S+G trials and 8.0% in the G trials.

The strict vector extrapolation model is no explanation for the extension of manual pointing.



(a) Perspective of the OI, GFP/dom



(b) Alternative perspective, IFP

Figure 6.8: *These Gesture Space Volumes are augmented to show the referential space of the pointing gestures of an individual DG in the S+G trials. The referential space is predicted by the vector extrapolation model, using GFP/dom (top) and IFP (bottom).*

6.3.2 The Pointing Cone Model

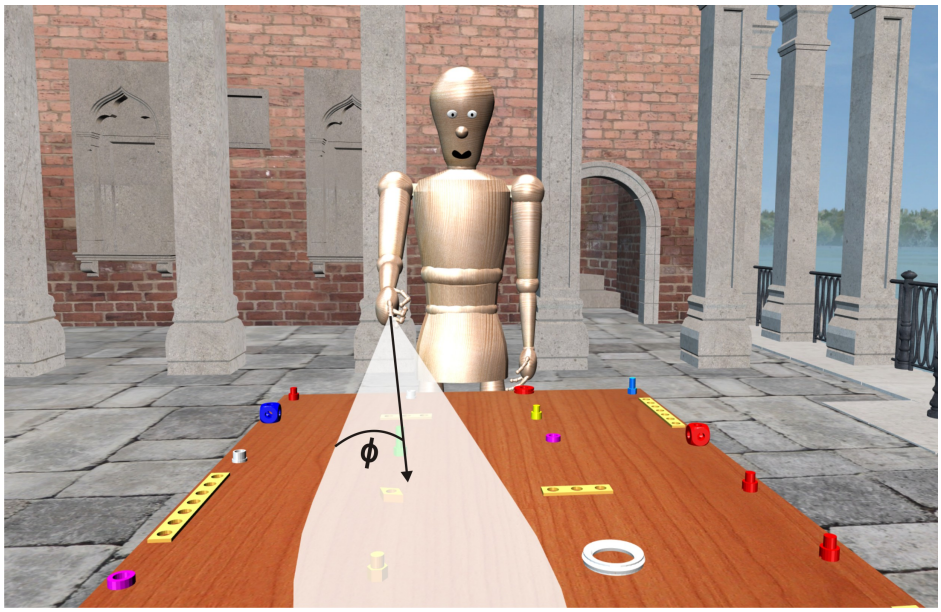


Figure 6.9: *The pointing cone model for pointing is parameterized by the origin (here the tip of the pointing finger), the direction (here GFP/dom) and an angle defining the aperture of the cone.*

A better candidate to model the extension of pointing is the pointing cone (see Figure 6.9), one of the shape-based dereferencing models. Pointing cone models try to take the increasing ambiguity of pointing into account. The equation for the pointing cone (Equation 3.9 on page 55) can be refined for GFP/dom in analogy to the vector extrapolation above:

$$\begin{aligned} G_{cone} : 0 &\geq \vec{y} \cdot \vec{v}_{GFP/dom} - |\vec{y}||\vec{v}_{GFP/dom}| \cos\phi \\ \text{with } \vec{y} &= \vec{x} - \vec{o}_{tip} \end{aligned} \tag{6.2}$$

The equation of the pointing cone model requires an additional parameter ϕ, which is half of the aperture angle of the cone. If ϕ is 0, the pointing cone model is identical to the vector extrapolation model. The aperture angle should be small enough to exclude false positives, i.e. objects that are mistakenly identified as referents. It should also be large enough, that as many referents as possible are correctly identified. In the following, an optimal ϕ for the manual pointing study will be approximated based on the recorded data.

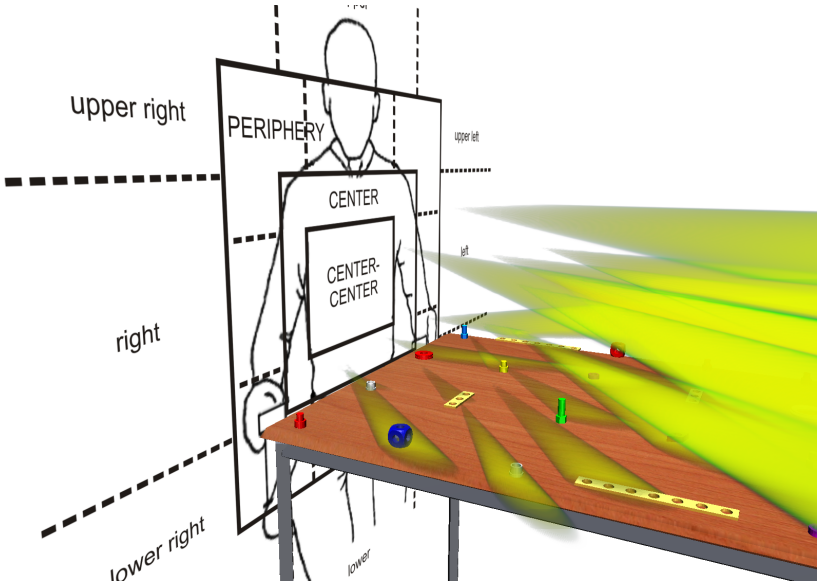
Finding the Optimal Aperture Angle To this ends, simulations were run with aperture angles between 0° and 100° ($\phi \in \{0° \cdots 50°\}$) to find the optimal angle for which the pointing cone model identifies most referents. In previous simulation runs to estimate the optimal angle of the pointing cone model the same data was used. In these simulations, distance errors were only measured based on the grid layout, which turned out to be an incomplete measure. These tests only probed whether the target referent was within the range of the pointing cone and whether the angular error was smaller than the minimal angles to the next row (see Table 4.1 on page 72). These preliminary results are published in Kranstedt (2007). However, as it turned out, these tests abstracted away too much from the exact topology of the neighboring objects. They did not provide a valid description of the performance of the pointing cone model, as they included too many false positives.

The simulation runs for the following analysis tested the intersections of the pointing cone models with the full set of objects in the pointing domain. In addition to the strict test which requires the referent object to be the one and only object within the cone, a pointing cone model with an *additional weighting function* is used. By use of the weighting function, the objects intersected by the cone are ordered according to their angular deviation from the axis of the cone. Objects closer to the central axis are ranked higher. Equation 6.3 shows the distance-based weighting, with 0 being the highest rank. An increasing distance will lead to more negative weights and thus a lower ranking.

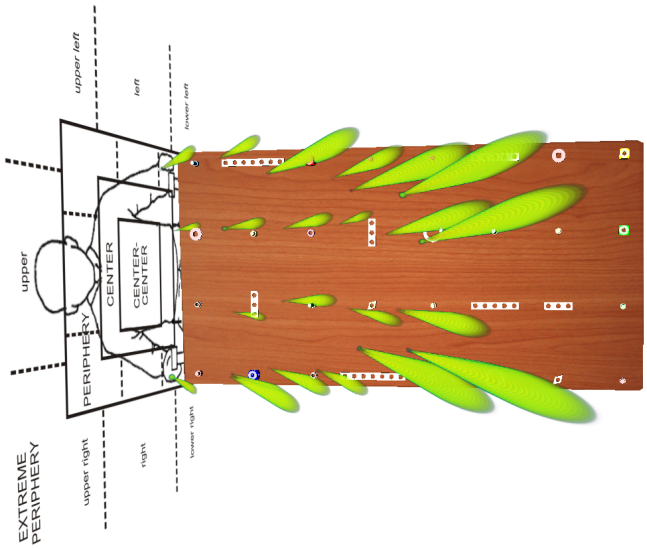
$$W_{\text{ortho}} \quad : \quad \mathscr{P} \times \mathscr{D} \rightarrow \mathscr{R} \tag{6.3}$$

$$W_{\text{ortho}}(p,r) \quad = \quad -\frac{|(\vec{G(r)} - \vec{o}_{tip}) \times (\vec{G(r)} - (\vec{o}_{tip} + \vec{v}_{GFP/dom}))|}{|\vec{v}_{GFP/dom}|} \tag{6.4}$$

As an illustration, the pointing cones for participant 04 of the manual pointing study are depicted in Figure 6.10.



(a) S+G Trials



(b) G Trials

Figure 6.10: *Pointing cones for participant 04 of the manual pointing study. The cones are primarily visible when pointing to distant objects in the S+G trials. In the G trials, the participant leaned forward and reduced the distance to the referent. In these cases, the cone model has problems similar to those of the vector extrapolation model.*

The simulations were run for the S+G trials and the G trials with similar results. As the accuracy of the GFP/dom model for the pointing direction is better for the S+G trials than for the G trials, the success rates of the pointing cones are also better in the simulations for the S+G trials than for the G trials. In addition, the human object identifiers only had problems dereferencing manual pointing gestures during the G trials. This makes the G trials more interesting when comparing the performance between the pointing model and the human. Therefore, the presentation in the following will concentrate on the G trials. The results of the simulation runs are categorized as follows:

unique the target referent object is the one and only object intersecting the cone; this is the strictest test

inference the target referent intersects the cone among other objects, but the target referent is ranked highest

failure the target referent intersects the cone among other objects, but a different object is ranked highest

clear failure some objects intersect the cone but the target referent is not among them

Figure 6.11 shows the simulation runs for the G trials. The data has been split into pointing acts that were directed at referents in the proximal area with a distance below or equal to 40 cm, and into pointing acts that were directed to referents beyond 40 cm. Referent objects in all rows of the pointing domain were considered.

Proximal area In the proximal area, a pointing cone with a ϕ of 13° uniquely identified the most referent objects (20.9%). This is about twice as many as were identified by the vector extrapolation model. With the same angle, an additional 7.8% were identified based on the weighting heuristics. At the same time, the weighting heuristics also falsely identified 6.7% objects that were not the intended referent, although the referent lay in the same cone. About 8.4% of the objects were falsely identified as a referent while the intended referent did not lie within the cone. The results also attest that the weighting function is important when dereferencing using the pointing cone. With a ϕ of 60°, the inference over the weighting function is able to successfully identify 58.2% of the referent objects. At the same time, however, 35.7% failed with the intended referent object inside the cone, and 3.9% failed completely. The human object identifier was able to correctly identify all referent objects (100%).

Distal area In the distal area, a pointing cone with a more narrow angle of $\phi = 7°$ uniquely identified the most referent objects (20.9%). Also, more identifications than in the proximal area were successfully achieved based on inference (14.1%). This went along with an increase in false positives (17.2%) and clear failures (17.8%). The results again attest the importance of the weighting function. With a ϕ of $60°$, the inference over the weighting function is able to successfully identify 44.8% of the referent objects. This is close to the performance of the human object identifier, who was able to correctly identify 56.4% of the referent objects in the distal area. The number of false positives at $\phi = 60°$ exceeded 50% (55.2%), but the intended referent object was always within the pointing cone.

Interpretation of the results The repeated simulations of 1037 pointing acts from the corpus on manual pointing have provided the optimal parameterization of the pointing cone model for the target scenario. For the *proximal area* an aperture angle of $26°$ ($\phi = 13°$) and for the *distal area* an aperture angle of $14°$ ($\phi = 7°$) should be chosen to optain the maximum number of unique identifications. This might be relevant in some application contexts to play safe.

With these settings, the pointing cone models perform much better than the vector extrapolation model. In the *proximal area*, the performance of the perfect human object identifier could not nearly be achieved. However, by applying the weighting function and increasing the aperture angle to $120°$ ($\phi = 60°$) the inference mechanism can be optimized to identify more than 50% percent of the referents.

In the distal area, the performance of the pointing cone model is slightly better than in the proximal area, regarding the absolute number of successes. At the same time, the number of false identifications nearly doubles. Compared to the human, the pointing cone model can win ground, but only because the performance of the human drops down to 56.4%.

- *Proximal area*: the pointing cone model is better than the vector extrapolation model, but not a good approximation of the performance of the human OI.
- *Distal area*: performance of the OIs decreases, but the pointing cone model maintains most of its discriminating power and thus achieves 79.4% of the OIs' performance.
- *Narrow aperture angle*: good to obtain a reasonable number of unique identifications.
- *Wide aperture angle and weighting heuristics*: the pointing cone model can identify about half of the referents, but at the same time a comparable number of false positives will be found.

How bad are failures? Along with the increased number of correct identifications comes an increase in false positives and clear failures when using the weighting function. Thus, a correct identification is only above chance if the pointing gesture is not interpreted detached from any context or from other restrictors on the referent, which might be part of a multimodal expression. In the cases marked as failures, the intended referent still lies within the pointing cone, it is just not the one object nearest to the axis of the cone, which is required by the weighting function. In these cases, the set of objects lying within the pointing cone is much smaller than the set of possible referents in the domain. The pointing cone can still provide valuable information if other restrictors apply, and can eliminate the false positives from the set a posteriori, for example in co-verbal manual pointing.

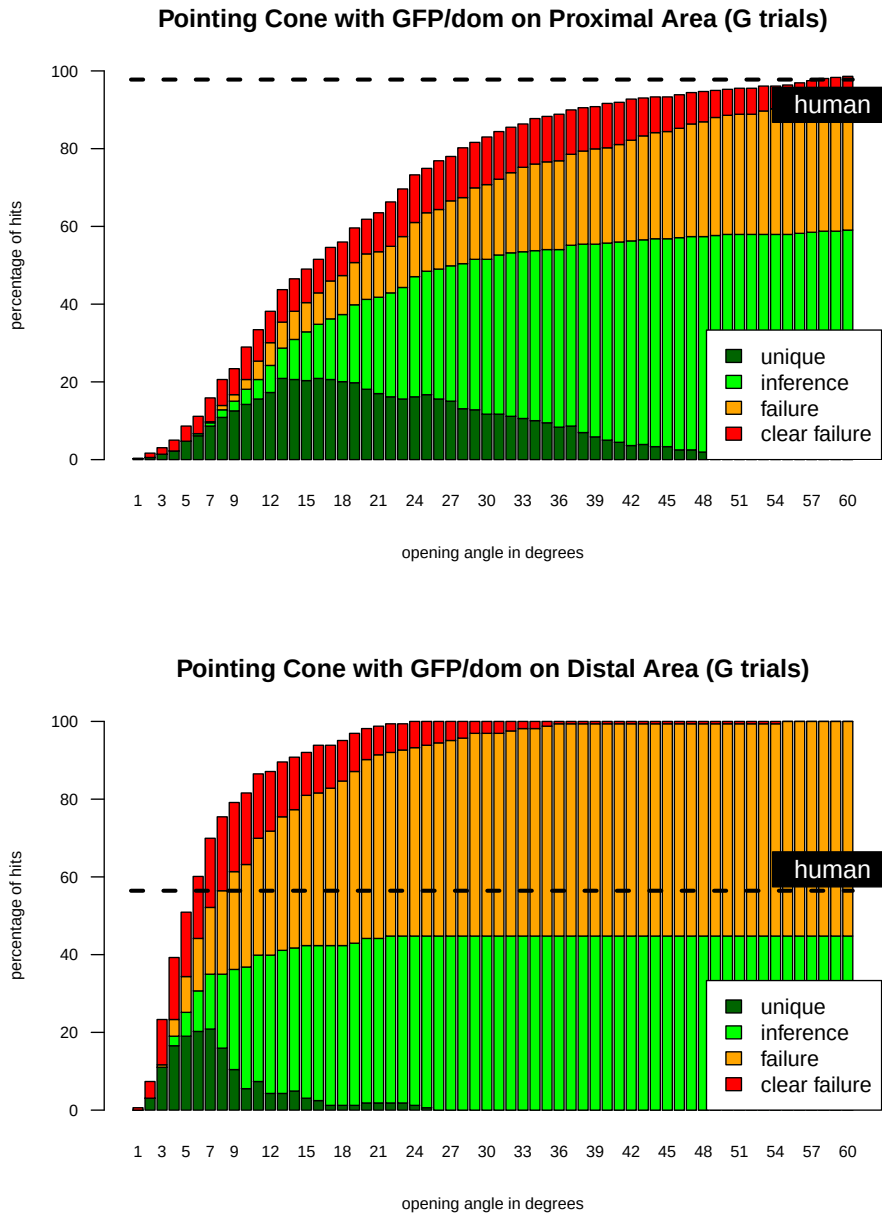


Figure 6.11: *Results of the simulation runs with GFP/dom pointing cone models, parameterized with different angles (top = proximal area, bottom = distal area).*

6.3.3 A Hybrid Pointing Model

The models for the extension of pointing evaluated so far did not provide satisfactory success rates. However, the evaluations provided some valuable insights.

1. If the weighting function is applied, an increase of the aperture angle will also increase the successful identifications.
2. If the aperture angle is increased, the pointing cone loses its identity and fewer referents are uniquely identified. In the end, the referent is primarily identified by the weighting function and not by the cone.
3. Most of the deictic references that cannot be identified using the pointing cone model are those to referents in the proximal area. This can intuitively be read from Figure 6.7: the mean angular error in the proximal area is much greater than the optimal aperture angles for the pointing cone model. Consequently, the narrow pointing cones will lead to more errors in this area.

On the other hand, the same diagram (Figure 6.7) also provides valuable information about the weighting function: the orthogonal errors will increase along with the increasing distance from the referent to the finger tip of the DG. In the small setting used for the study, this does not have much consequence on the performance of the weighting function, which is based on the orthogonal distance. The mean error at the distance of 1 m is about 10 cm, which is still enough to discriminate the referent in most cases. The DGs' pointing hand will be hovering about the same height above the table when pointing to distant objects and thus already maintain at least a distance of 10 cm from most other objects. In other scenarios this would be different. If the domain of possible referents contains objects that are distributed over larger distances, the weighting function based on the orthogonal error will not be able to "get around" objects closer to the pointing hand.

> The pointing cone model based on angular errors works best for precise pointing and distant referents. The weighting function based on orthogonal errors works best when pointing to proximal referents.

Considering these insights, the optimal pointing model would combine the properties of both approaches: it would use orthogonal errors in the proximity and angular errors in the distance. Such a hybrid model is specified in Equation 6.5.

$$G_{hybrid} : \begin{cases} G_{cone} & \text{if } |\vec{o}_{tip} - G(r)| \geq d_{thresh} \\ G_{cone} \cup |\vec{o}_{tip} - G(r)| < d_{max} & \text{if } |\vec{o}_{tip} - G(r)| < d_{thresh} \end{cases} \quad (6.5)$$

The arm length of the extended pointing arm can be used as threshold d_{thresh} to differentiate between the proximal and distal region. The performance of this model has again been tested on the data from the manual pointing study. Figure 6.12 shows the simulation runs for the G trials, again split into proximal and distal areas.

In the proximal and distal area, the data for the unique identifications are equal to those obtained by the basic pointing cone model, as this component of the hybrid model was left unchanged. The relevant differences show up when the orthogonal errors are considered.

In the *proximal area*, the orthogonal errors provide a basic rate of success of 66.0%. As this is not a function of the opening angle of the cone, this contribution of the orthogonal error to the success rate remains constant. Even in the proximal area, the pointing cone model can increase the success rate to 89.4% ($\phi = 69°$). If $\phi = 13°$, the combined success rate of unique and inferential identifications is at 76.3% (22.3% unique and 54% inference).

In the *distal area*, the hybrid model achieves a higher success rate than the human object identifier. The basic success rate contributed by the orthogonal errors is 57.7% and the human OIs only identified 56.4%. The success rate is further improved to 62% by the contribution of the pointing cone.

The created hybrid pointing model considers the distinction of proximal and distal pointing by combining orthogonal and angular measurements. It provides the best performance on the data from the manual pointing study.

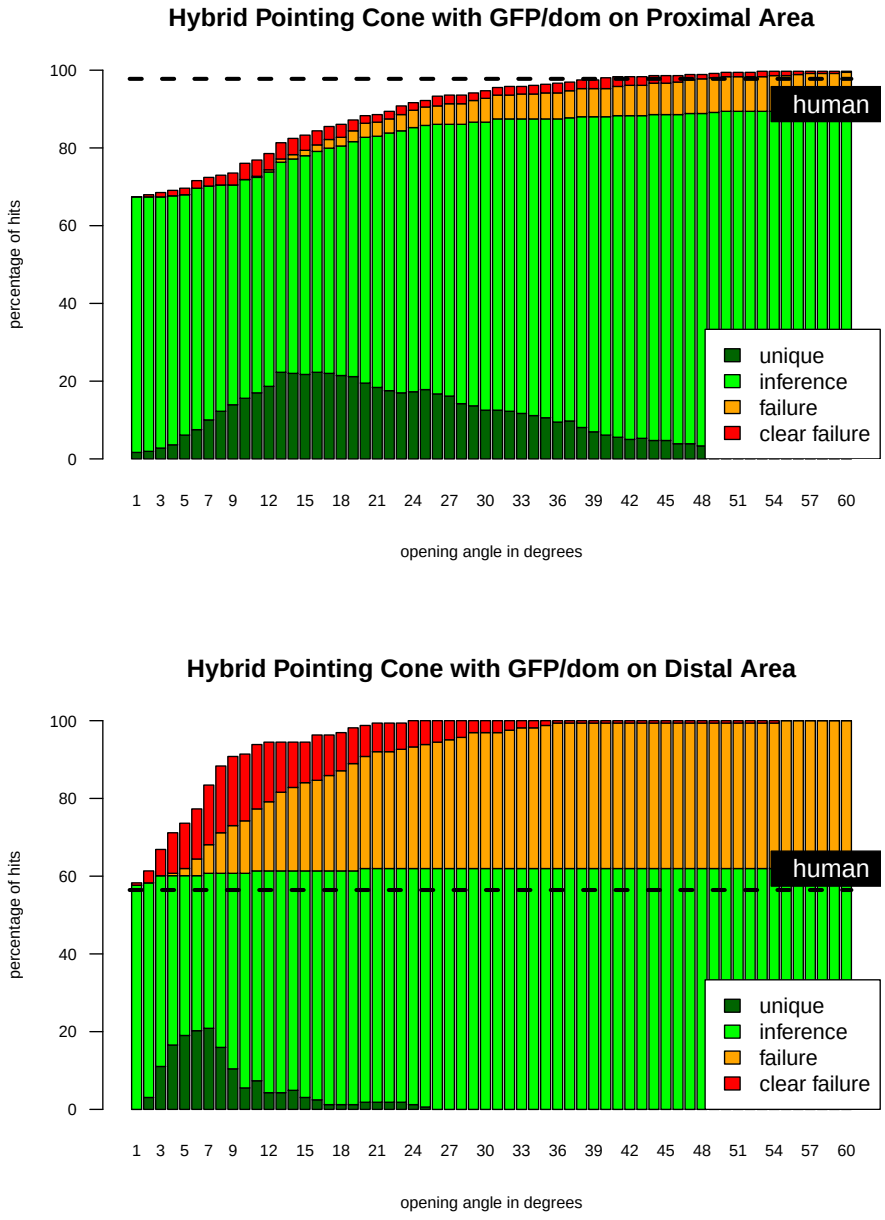


Figure 6.12: *Results of the simulation runs with the hybrid pointing cone models, with a direction predicted by GFP/dom.*

6.4 Modeling Gaze Pointing

With the hybrid model developed in the previous section, an accurate model of the extension of manual pointing has been found. This section now turns towards pointing models for *gaze pointing*. The two studies on direction-based and location-based gaze pointing presented in Chapter 5 have already testified that gaze pointing is accurate and precise, especially when compared to manual pointing. This section thus only refines the pointing model for *location-based* gaze pointing. While the technology to detect the point of regard of eye gaze in 3D has been developed in the context of deictic reference, it can be used to assess more basic processes, such as the *flow of visual attention* in space. This section presents a new technique to visualize the volumes of attention, which can be identified in this way.

6.4.1 Attention Volumes

Attention maps or heatmaps were introduced in Section 2.4.6 for 2D surfaces. Attention Volumes extend these concepts to 3D space. Figure 6.13 shows the Attention Volume equivalent to the 3D scanpath depicted in Figure 5.12. Similar to the 3D scanpaths, individual points of regard are depicted in Attention Volumes. The sequence between individual points of regard is not visualized. Attention Volumes focus more on aggregating information over several fixations and/or participants than on individual scanpaths. Looking closer at the visualizations of the points of regard one can recognize that they are not depicted as a solid geometry as in the 3D scanpaths, but as a color gradient. This gradient represents the likely distribution of attention based on the assumption of the cone of attention defined by the area of high acuity inside the visual field, as explained in Section 5.2.1. The distribution of the colors ranges from red for areas of a high probability of attention, over green and yellow to transparent unshaded areas where no attention has been registered.

When aggregating over several participants, the distributions can be superimposed and normalized to generate a visualization of the likely overall distribution of attention. This has been done for the gaze pointing data recorded in the study on location-based gaze pointing to create Figure 6.14, similar to the 3D scanpaths shown in Figure 5.14.

6.4.2 A Model for Gaze Extension

The point of regard in 3D is approximated in the first instance by the shape of a sphere (see G_{gaze} in Equation 6.6). This sphere is accompanied by a weighting function W_{gaze}, which models a gaussian distribution around the center of the point of regard in 3D. This distribution is slightly distorted by taking the opening angle of the area of high visual acuity into account.

$$G_{gaze} : 0 = \vec{y} \cdot \vec{v} - |\vec{y}||\vec{v}| \cos\phi \tag{6.6}$$

$$W_{gaze}(\vec{x}) = d(t) e^{-\frac{|\vec{x}-\vec{p}_{por}|^2}{\sigma(\vec{p}_{eye},\vec{x})}} \tag{6.7}$$

with $\vec{p}_{por}$: 3D point of regard

$\vec{p}_{eye}$: position of the eye

$d(t)$: amplification factor depending on the duration

Examples of predictions of the updated model for gaze pointing are presented in Figure 6.15. The amplification factor $d(t)$ amplifies the distribution depending on the duration of the fixation. Longer durations will lead to higher amplitudes of the gaussian function, which will be visualized by a darker shading.

If this function is used during on-line interpretation of gaze pointing, a threshold can be used to detect the fixated area. The referent object can then be identified by intersecting the volume with the domain of possible referents. Possible thresholds have been specified in Section 2.4.4; a typical threshold is 250 ms. If the function is used on-line, the volume has to be updated regularly to fade out older fixations. Otherwise the distributions will cumulate until everything will be interpreted as being attended to.

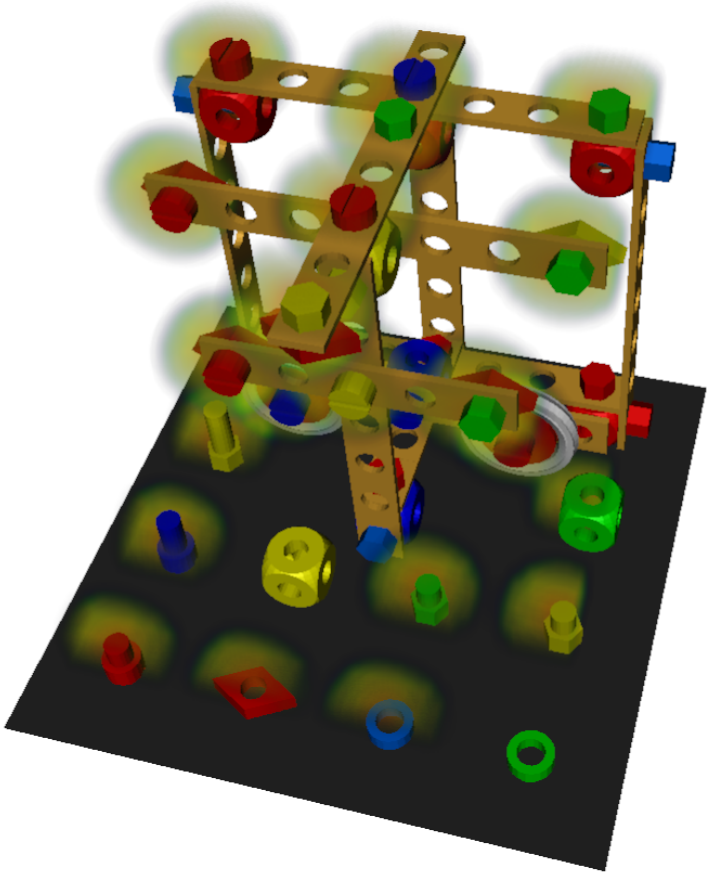
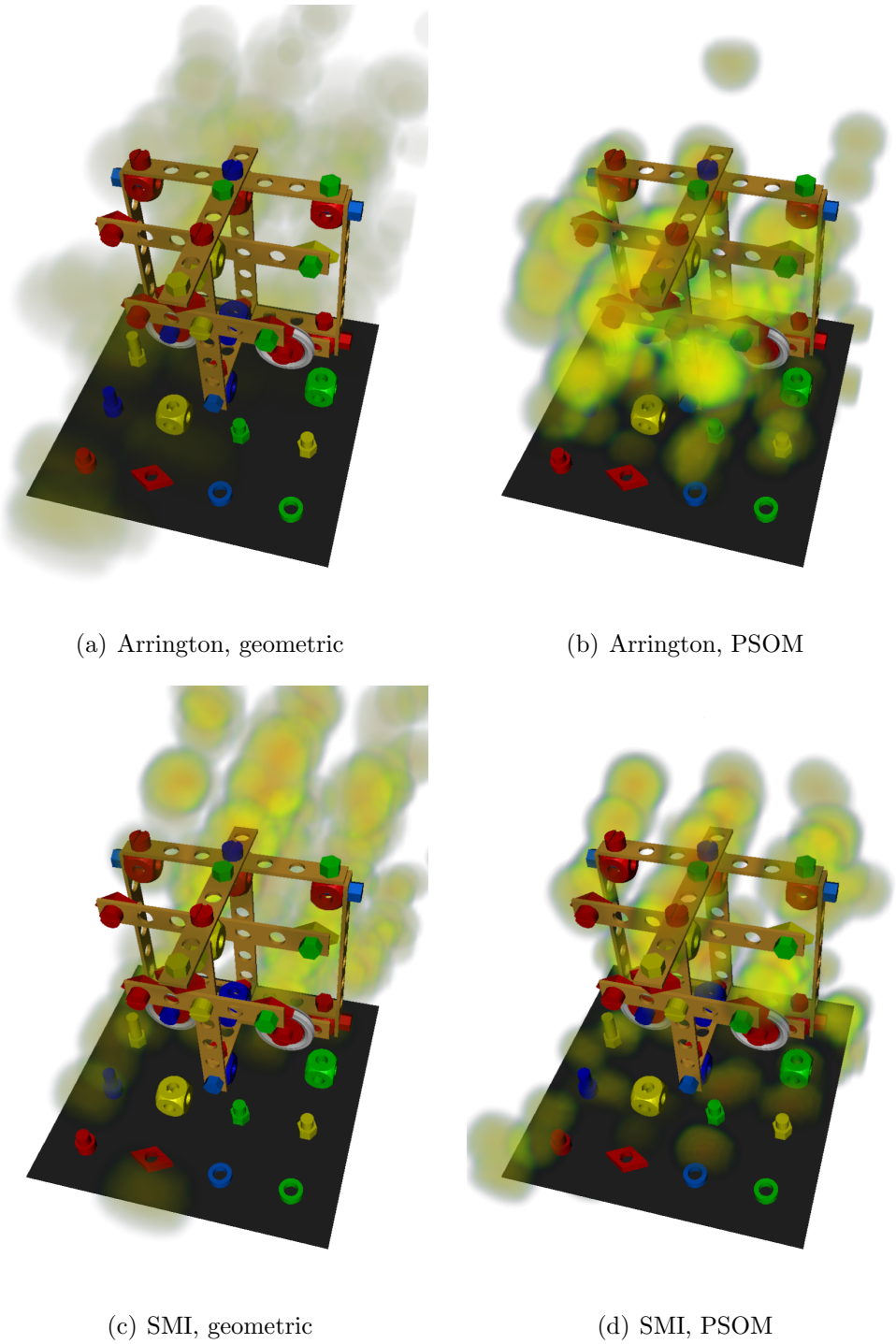


Figure 6.13: *Attention volumes show the distribution of visual attention, here demonstrated using the example of the purely hypothetical optimal target attention distribution for the target objects presented in the study on estimating the 3D points of regard (Section 5.6). Detected points of regard are visualized as color distributions from red fading out over yellow and green to transparent. Areas of red color are more likely to have received attention than yellow, green or unshaded areas.*



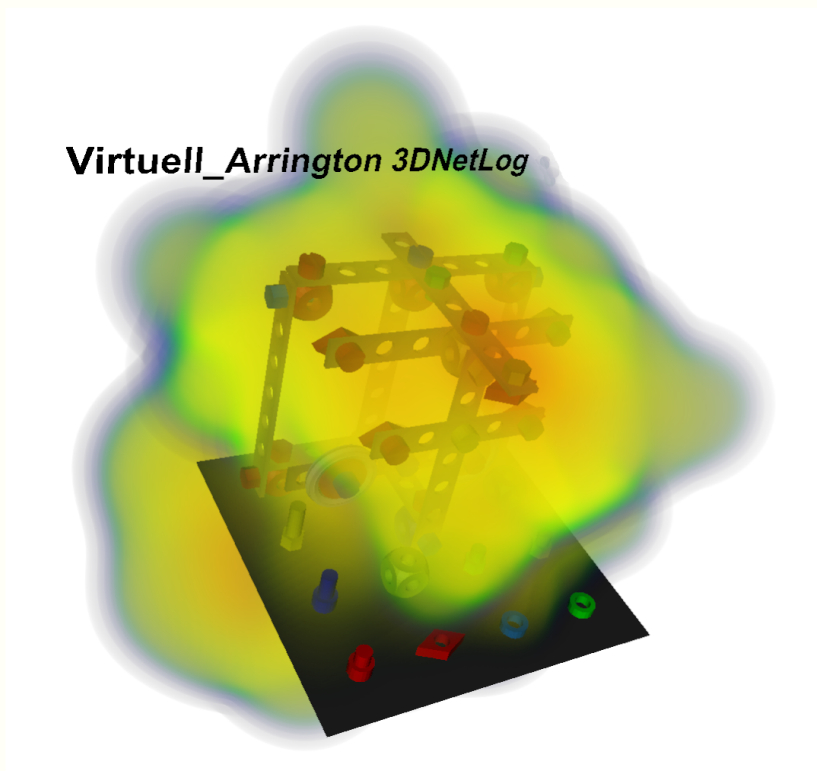
(a) Arrington, geometric

(b) Arrington, PSOM

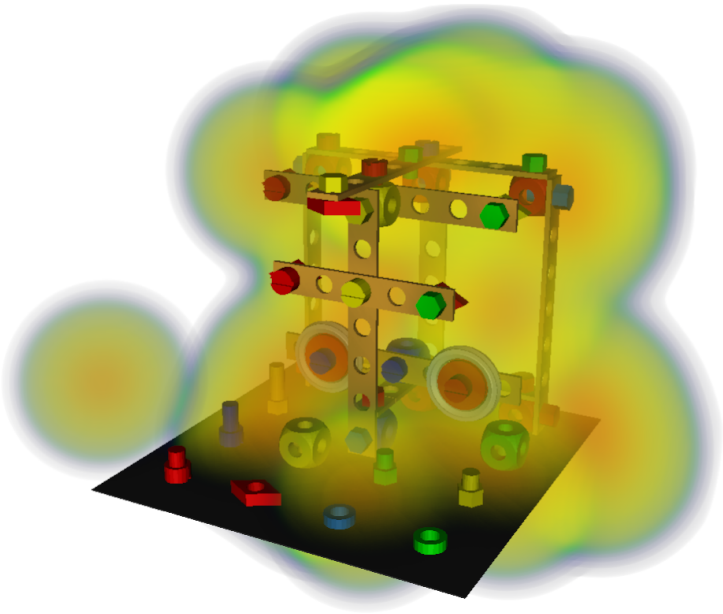
(c) SMI, geometric

(d) SMI, PSOM

Figure 6.14: *Attention Volumes of the data collected in the study on location-based gaze pointing (Section 5.6). Compared to the 3D scanpaths over all participants (Figure 5.14), the underlying Baufix assembly is still visible, and the intensity and the colors of the gradients reflect the amount of attention the area has received.*



(a) Arrington Research, PSOM



(b) SMI, PSOM

Figure 6.15: *The Attention Volumes presented here show the distribution of attention as predicted by the updated gaze pointing model. The new model takes the distance of the point of regard from the eye into account and modifies the width of the gaussian distribution according to the angle of high visual acuity.*

6.5 Integrating Pointing Models with a Conversational Interface

This section gives an account of the way the developed models for pointing can be integrated in a conversational interface. First, the DRIVE framework is described. DRIVE uses modality-specific models to derive the potential referents of a pointing gesture. Its input are the raw sensor data provided by the different tracking systems. Its output are weighted lists of potential referents. The next step then is the multimodal integration, which is done by a multimodal reference resolution engine. Both steps are described in the following sections.

6.5.1 Deictic Reference in Virtual Environments

The models that have been developed in this thesis, especially those described in this chapter, have been integrated in a component-based framework for deictic reference called D*eictic* R*eference* I*n* V*irtual* E*nvironments* (DRIVE). In this section, only a brief overview of DRIVE can be given. An extensive description of DRIVE can be found in Chapter A in the Appendix.

The DRIVE framework is based on X3D (ISO 19775-1:2004, 2004) and the extensions provided by instant**reality** (Fellner, Behr & Bockholt, 2009) and InstantIO developed by Fraunhofer IGD, Germany. In the DRIVE framework, functional components, the nodes, are connected to a data-flow network via routes. There are nodes for interfacing the tracking systems, such as the IO::EyeTracker node and the IO::ARTpro node (see Section A.2.2 and Section A.2.3) as well as computational nodes, e.g., to detect fixations (see Section A.3.1). These nodes are interconnected to complex networks for detecting pointing gestures and for deriving relevant features of the gestures. Finally, once a pointing gesture has been detected, a set of dereferencing nodes applies the appropriate pointing models developed in this thesis and retrieves the potential referents, if there are any.

The components of the DRIVE framework are thus at the core of analyzing and interpreting gaze and manual pointing in a staged process. In the context of the interpretation of multimodal place deixis, the interconnected components of DRIVE provide a rated set of possible referents for either gaze or manual pointing as a result. The rating is thereby done according to the weighting functions described above.

```
(and  (instance "?object-1" ontology:coar:baufix:BLOCK)
      (instance "?object-2" ontology:coar:baufix:BLOCK)
      (has-color "?object-1" RED)
      (very (is-target-of-manual-pointing "?object-1" t1))
      (very (is-target-of-manual-pointing "?object-2" t2))
      (prefer (is-target-of-gaze-pointing "?object-1" t1))
      (prefer (is-target-of-gaze-pointing "?object-2" t2)))
```

Listing 6.1: *The constraint satisfaction problem for the instruction "put this red block on this block".*

The components provided by DRIVE can also be used to realize other aspects of communication. The components for interpreting eye-tracking data have so far been used to realize turn-taking, check-backs and high-level communication functions such as shared attention and joint attention Pfeiffer-Lessmann & Wachsmuth (2008) (see Section 7.1.2).

6.5.2 Multimodal Reference Resolution

The logical next step is the integration of this multimodal information. One possible approach to this problem, which has been followed by Pfeiffer & Latoschik (2004) in the contexts of the CRC 360 and the DFG project Virtuelle Werkstatt (Virtual Workshop), is to define the multimodal reference resolution process as a fuzzy constraint satisfaction problem (fCSP). This is illustrated in the following with a small example.

An instruction such as "put this red block on this block" would result in the specification of the constraint satisfaction problem shown in Listing 6.1. The listing in the description language of the fCSP describes a graph of variables, **?object-1** and **?object-2**, which can represent any object in the domain of possible referents. The number of variables that are instantiated in the graph depends in this example on the number of noun-phrases. The type of variables in this case is determined by the specific noun which has been used. Further information that is parsed from the verbal instruction, such as the color of the first object, is attached to the variables as constraints (e.g. **has-color**).

In addition to the constraints derived from speech, several other constraints can be added, for details see Pfeiffer & Latoschik (2004) (or the full description in German in Pfeiffer (2003)). Examples are constraints that express certain preferences, such as that objects closer to the interlocutors will be more

likely to be the target of a manipulation, or that objects which have recently been the target of an action will be preferred over others. Relevant in this context are the constraints that are added to integrate information from gaze and manual pointing. The constraint **is-target-of-manual-pointing**, for example, evaluates possible variable assignments based on the rated extension set provided by DRIVE. The approximate moment in time at which the pointing gesture occurred is provided as an additional parameter, which in turn is derived from the time the associate word has been uttered in the speech channel.

The default proceeding of the reference resolution component is always to add constraints for gaze or manual pointing whenever a new reference has been identified in speech, so that possible referents that have been gazed at or manually pointed to are preferred over others in the current context. At this stage it is unknown whether a pointing gesture was actually made. In the fuzzy CSP, this uncertainty is modeled by the operator **prefer**, which marks that the following constraint should be applied if possible, but, if there has been no pointing gesture, the results are not affected. In the example of Listing 6.1, this is the case for gaze pointing. For manual pointing, the reference resolution component uses a different operator, **very**, because the user used "this" in the speech channel. This could be taken as a signal for a manual pointing gesture, and thus the harder fuzzy operator **very** is used instead of the softer operator **prefer**. The reference resolution component expects that the deictic pronoun goes along with a clarifying manual pointing gesture.

In the example provided above, the pointing models and their implementation in DRIVE allowed the reference resolution system to understand multimodal deictic expressions where formerly only verbal expressions were understood. The basic interface between DRIVE and higher-level processes for multimodal integration is small and concise. It consists primarily of a data exchange of an ordered weighted list of possible referents for a specific moment in time. The component-based architecture of DRIVE is versatile enough to be adopted to other systems as well.

6.6 Summary

This chapter presented a thorough investigation of the pointing studies to derive models for the extension of pointing. Regarding manual pointing, it identified the need for reconsidering the original method of analyzing the data

from the corpus on manual pointing. Using the distances between the finger tip and the intended referent as measurement, categorial answers were given on the differences in accuracy of different models for the direction of pointing (see Section 6.2). This paved the way for the development of the GFP/dom model for the direction of gaze pointing, which takes the dynamically switching dominance of an individual eye into account. This new GFP/dom model predicted the direction of pointing with a previously unachieved accuracy.

Based on the *GFP/dom* model, different models of the extension of pointing were tested subsequently in Section 6.3. The well-known vector extrapolation model and the pointing cone model alone did not provide satisfying results. A comparison of the strengths and weaknesses of the different approaches, and an analysis of the interplay between the different pointing cones and weighting functions led to the development of a new pointing model. The key factor of this hybrid pointing model is the recognition of the differences in proximal and distal pointing. The hybrid model finally achieves a success rate of 76.3% in the proximal area and 62% in the distal area, which exceeds the performance of the human object identifiers in the same area.

Gaze pointing had already been identified as being very accurate, especially when compared to manual pointing. The model of location-based gaze pointing was refined further in Section 6.4. The updated model takes the distance of the point of regard and the eye into account to estimate the probability of the visual attention in 3D space. To visualize the results of this updated model and to reveal the flow of visual attention in 3D, a new visualization technique called *Attention Volumes* has been introduced.

Finally, the integration of the pointing models in a conversational interface was demonstrated in Section 6.5, where the DRIVE framework as well as the multimodal integration using a fCSP-based reference resolution were presented. This section also provided an example, which showed how gaze and manual pointing can be integrated with speech. Applications which make use of the models presented in this chapter are reviewed in the following chapter.

Chapter 7

Applications and Conclusion

This chapter demonstrates applications of the pointing models, the interaction framework DRIVE (D*eictic* R*eference* I*n* V*irtual* E*nvironments*) and the technologies for 3D gaze tracking which have been developed in this thesis. These applications are selected to highlight different aspects of this work and to emphasize the transferability of the achievements. The résumé distills the results on modeling pointing, discusses their implications for conversational interfaces and concludes this thesis.

7.1 Applications with DRIVE

The DRIVE framework (see Chapter A) and its prototypes have been used in several research projects and applications. First of all, the I*nteractive* A*ugmented* D*ata* E*xplorer* (IADE) (see Section 4.6 and Section 4.8) for investigating multimodal interaction uses DRIVE in its simulations and during interactive sessions to allow the researcher to interact with the objects of investigation as well as the controls of IADE's user interface.

7.1.1 DRIVE for Processing Multimodal Expressions

The work on understanding gaze and manual pointing presented in this thesis started within the scope of the *Collaborative Research Centre 360: Situated Artificial Communicators*. A central application to demonstrate the research results is the *Virtual Constructor* (Jung, Hoffhenke & Wachsmuth, 1998), a system for rapid prototyping of construction processes using natural

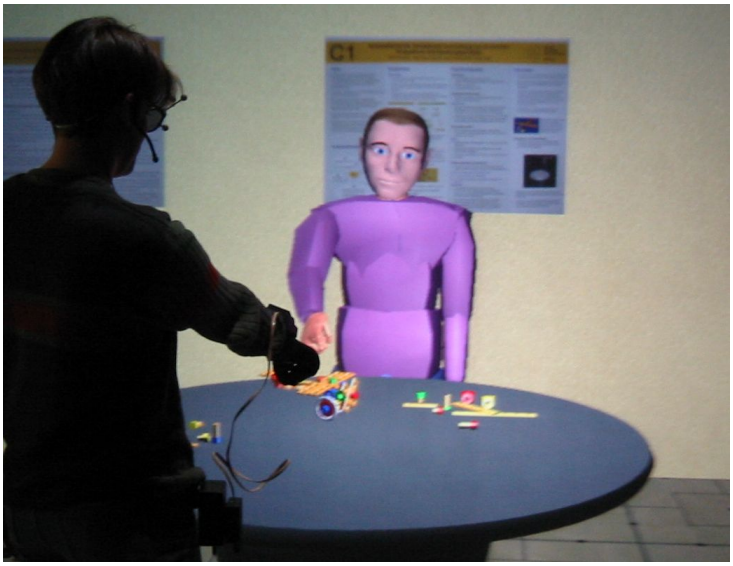


Figure 7.1: *In the Virtual Constructor, users can interact with the system, which is represented by the embodied conversational agent Max, to construct aggregates from a virtual toolkit. The system is able to understand multimodal deictic expressions to refer to objects in this immersive virtual environment.*

speech and gestures (see Figure 7.1). In this multi-agent system, deictic references are resolved using an agent specialized for reference resolution. The most advanced reference resolution agent of the Virtual Constructor models the dereferencing problem of multimodal expressions using fuzzy constraint satisfaction problems (Pfeiffer, 2003; Pfeiffer, Voss & Latoschik, 2003; Pfeiffer & Latoschik, 2004).

DRIVE detects gaze and manual pointing gestures of the user in real-time, and dereferences them using the pointing models developed in Chapter 6. The manual pointing model developed in this thesis has been evaluated in Section 6.3.3 on a domain of possible referents consisting of the type of objects used in the Virtual Constructor. The hybrid pointing model that has been developed in this thesis outperforms the previous models in this domain (see Section 6.3). It provides high accuracy by modeling the direction of pointing using the new *GFP/dom* model, which takes the dominant eye of the user into account (see Section 6.2). The model also provides a high rate of success in identifying referents by using a pointing model that combines a pointing cone for precise pointing to distant referents with a weighting of possible referents in the proximal area, based on their orthogonal distance to the pointing ray. For gaze pointing, DRIVE implements the location-based approach to detect

the point of regard in 3D, which was evaluated in Section 5.6. Using this model, DRIVE provides improved capabilities for identifying referents even if they are partially occluded.

DRIVE provides histories (see Section A.5.7) of the identified referents to the reference resolution engine in form of a list of ranked candidates. These histories can be used in reference resolution to integrate gaze and manual pointing with constraints identified in the verbal part of the multimodal instruction and constraints derived from the dialog context. Section 6.5 provides a detailed description of how the results provided by DRIVE are formulated as constraints in the fCSP system, and finally of how the multimodal deictic references within an utterance can be interpreted.

The pointing models implemented in DRIVE provide accurate information about the referents of gaze as well as manual pointing for multimodal integration of place deixis within a conversational interface.

7.1.2 DRIVE for Embodied Conversational Agents

DRIVE has been used to support the work of Pfeiffer-Lessmann & Wachsmuth (2008) in the project A1, "Modelling Partners", of the Collaborative Research Center 673, "Alignment in Communication". In this project, a user interacts with the embodied conversational agent Max within an immersive virtual environment, powered by a TRI-SPACE virtual reality system. Max and his interlocutor work on a 3D model of a small town (see Figure 7.2 and the linked video). Using DRIVE, Pfeiffer-Lessmann & Wachsmuth (2008) made Max aware of the visual attention of his human interlocutors. They made use of this information in a Belief-Desire-Intention architecture to monitor turn-taking signals, check-backs, shared attention and to establish joint attention with the virtual human in cooperative dialogs.

For example, if Max wants to introduce a new object, he has multiple options: he can use a complex verbal expression, an effortful multimodal expression accompanied by a co-verbal manual pointing gesture, or just a short gaze at the intended object. Without further feedback from the user, such as check-backs, Max cannot know for sure whether the user is following his gaze. Following a safe-play strategy, Max may thus put much effort into an explicit reference.

DRIVE allows Max to follow the gaze of his interlocutor in real-time. To this end, the user wears a ViewPoint PC60 eye tracker from Arrington Research,

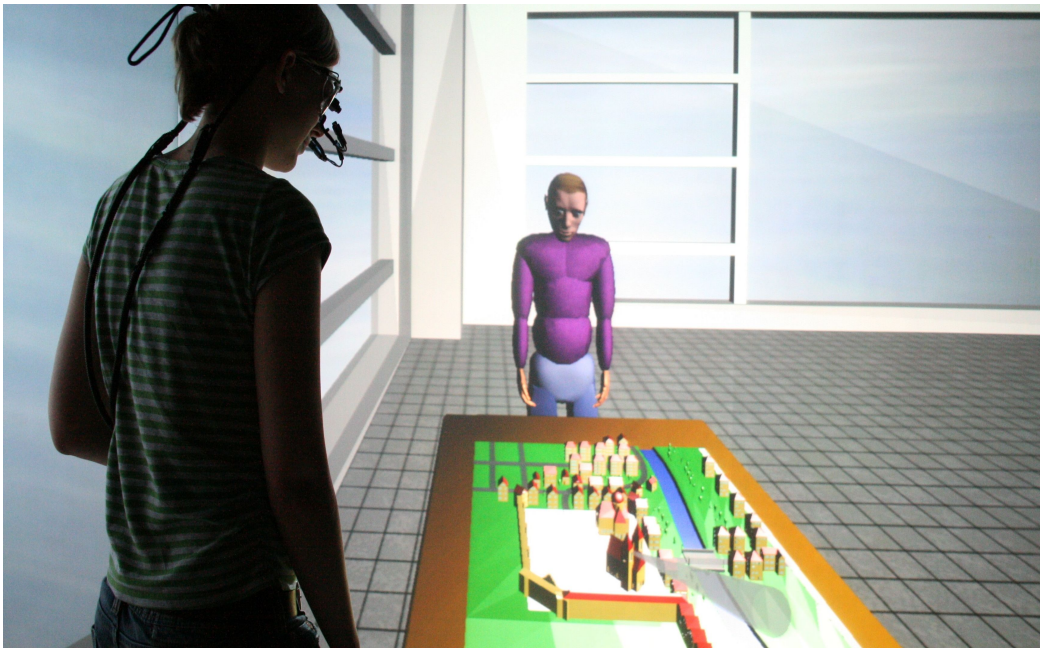


Figure 7.2: *Conversational interfaces can use gaze information to monitor attention and facilitate joint attention. The green cylinder pointing at the church visualizes the current pointing ray of the user's right eye (which narrows down due to the perspective projection). The cylinder appears displaced, as the current perspective has been corrected for the camera. Under normal conditions, this cylinder can only be rendered for the opposite eye (so the left eye sees the cylinder for the right eye), since otherwise the user would only see the bottom of the cylinder in front of each eye, due to the high accuracy of the system. In the PDF version, a video is linked that can be accessed by clicking at the photo.*

which was positively evaluated for this kind of interaction setting in the study presented in Section 5.1. DRIVE integrates the 2D gaze positions provided by the eye tracker with the 3D head position from a motion tracking system to construct the direction of gaze in 3D space (see Section A.3.3). If the binocular mode of the eye tracker is used, DRIVE can also provide the precise location of the point of regard in 3D space (see Section A.3.4) using the procedure developed and evaluated in Section 5.6. DRIVE further supports Max by provisioning a history of the objects the user has attended to during the last seconds (see Section A.5.7).

Supported by DRIVE, Max can now be informed of the attention of his interlocutor and can establish whether the interlocutor has correctly identified

the referent. This allows Max to use swift gaze pointing to introduce and refer to objects. If Max then notices that the interlocutor is not attending to the object in a certain time frame, Max can escalate his efforts, e.g. by making a manual pointing gesture or a full multimodal expression.

DRIVE enables embodied conversational agents to follow the visual attention of their interlocutors. As a consequence, the agents can use this information to plan their multimodal expressions more efficiently (e.g. skipping expressions as soon as the user attends to the relevant object). Overall, this contributes to a more natural appearance of the agent.

7.1.3 DRIVE for Attention-Aware Interfaces

Natural pointing via hand and gaze is not restricted to human-computer interaction with systems which understand natural language. SoN∀R (Social Networks in Virtual Reality) is an immersive 3D exploration tool for the social network Last.FM. SoN∀R has been developed in one of the student projects supervised by the author (Bluhm, Eickmeyer, Feith, Mattar & Pfeiffer, 2009).

SoN∀R is an interactive viewer for augmented graphs in immersive virtual reality (see Figure 7.3 and the linked video). It visualizes the social network of the user as a large universe of nodes. These nodes can represent users, music tracks and artists. Relationships between nodes, such as the music tracks produced by an artist or the artists liked by a user, are represented as links between the nodes. In this way, users, tracks and artists form a multidimensional graph that can be interactively explored by the user in real-time.

DRIVE allows the user to select nodes using pointing gestures (see Section A.5). These nodes can then be manipulated and the graph can be further expanded to visualize different relations. User interface components, such as menus or buttons, as well as auxiliary information augmenting the nodes (information about the artist, the year of the release, etc.) are only presented on demand. For this purpose, DRIVE provides information about the current visual attention of the user (see Section A.3.3), which allows SoN∀R to selectively fade in these augmentations in the local vicinity of the currently fixated node.

SoN∀R is an example par excellence for the hybrid model of manual pointing (see Section 6.3.3) and for the point of regard in 3D. In the 3D visualization of the network, the nodes are fully distributed across the proximal as well as the distal area. In addition, nodes in the proximal area will partly occlude

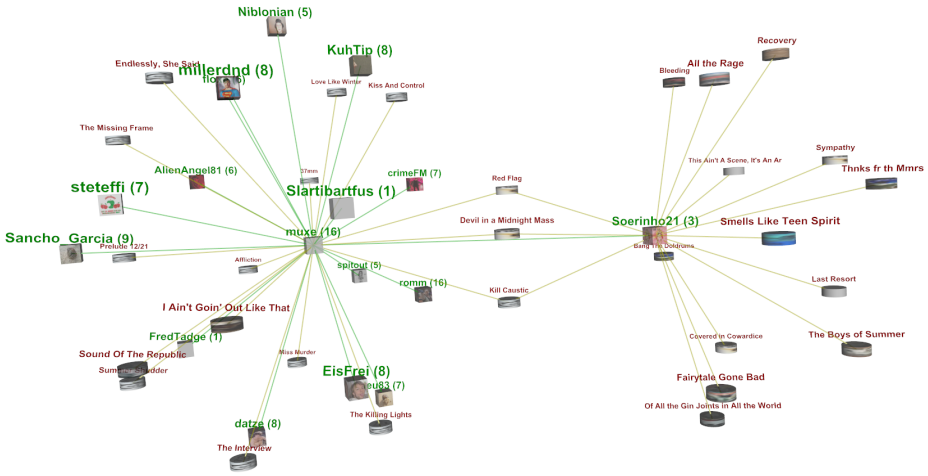


Figure 7.3: *The use of natural pointing gestures and gaze-based interaction extends to other interfaces and applications, such as SoN∀R , a tool for the immersive exploration of social networks.*

nodes in the distal area because of their great number. With the developed models, DRIVE allows the user to point arround the nodes in the proximal area, and to attend to and select nodes that are far away.

DRIVE enables interactive applications to react to pointing gestures. In information visualization, DRIVE can be used to reduce clutter in the user interface. DRIVE empowers the application to selectively present relevant information in the area of visual attention.

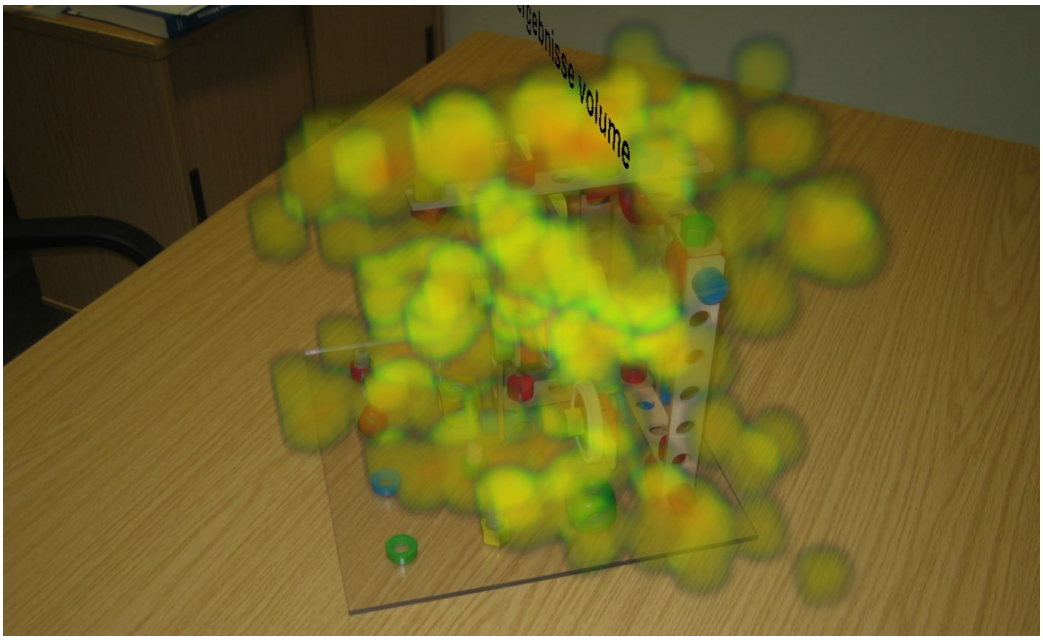
7.1.4 DRIVE in Real World Applications

Virtual reality has been an important enabling technology to drive the research of this thesis. The developed concepts and some of the technologies also extend to the real world. In the following section, an example will be given that demonstrates the application of the technology for tracking visual attention in 3D space, developed and evaluated in Chapter 5, in the real world.

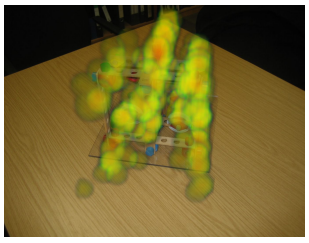
Figure 7.4 presents examples from a study where DRIVE was used to track the visual attention of participants on a real-world replication of the Baufix assembly used in one of the studies. Using the location-based gaze pointing model (see Section 6.4), DRIVE is able to estimate the 3D points of regard in real-time, without the need for a virtual representation of the scene. This is

new compared to other approaches to tracking attention in 3D, which either do not work in real-time (Mitsugami et al., 2003), or use direction-based models for gaze pointing (Duchowski et al., 2002). They therefore require an explicit modeling of all relevant geometries – also over time – to intersect with the pointing ray (see also Papenmeier & Huff (2010) for an offline analysis using dynamic 3D models).

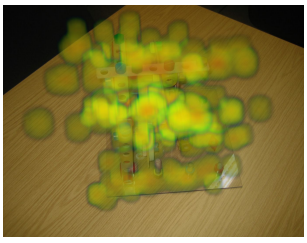
The examples in Figure 7.4 demonstrate that the developed visualizations, attention volumes (see Section 6.4.1) and 3D scanpaths (see Section 5.11.1), can be applied in the real world as well.



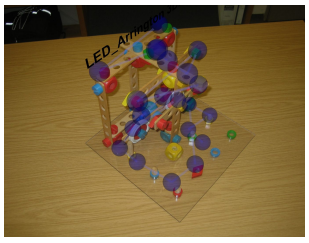
(a) Left view of an attention volume recorded on a real object (SMI/PSOM). In this study, data from 10 participants observing the real object was recorded.



(b) frontal view



(c) right view



(d) individual 3D scanpath

Figure 7.4: *Measuring the 3D point of regard on real objects. Shown are different perspectives of the target Baufix assembly.*

7.2 Résumé

This thesis covers the full cycle of scientific research, starting with basic research, the development of scientific methods, modeling and finally implementation and evaluation. In the following, the most important contributions are highlighted. The presentation is organized along three main pillars: basic research, scientific methods and human-computer interfaces. The latter includes the implementation and evaluation of the developed models in different application scenarios.

7.2.1 Contributions to Basic Research

Starting point of this thesis were the three fundamental questions about the *when*, *where* and *which* of multimodal deixis with gaze and gesture:

- When does the interlocutor perform a pointing gesture, and what is the relevant time interval of the whole gesture trajectory?
- Where does the interlocutor point to (direction)?
- Which object does the interlocutor refer to with the pointing gesture?

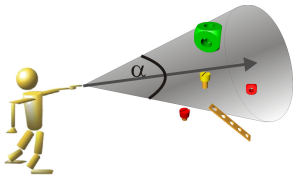
Answers to the *When* Question

Reviewing literature from linguistics and psychology (see Chapter 2), as well as on human-computer interaction (see Chapter 3), it was found that there is already a concise concept of the timing of manual and gaze pointing, which can be directly derived from the movements. Hand, arm and eye movements are visible and easily accessible to humans – otherwise pointing gestures would be indeed pointless. Hence, the timing of pointing gestures has already been addressed by scientific investigation employing methods such as video recordings or eye tracking. As a result of the literature review, the question on the *when* of pointing was considered as being answered. At the same time, however, it was found that there were no satisfying answers to the *where* and *which* questions and thus the thesis focussed on answering these.

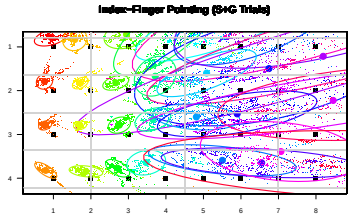
Answers to the *Where* Question

We gesture in the 3D space that surrounds us. Approaching the *where* question therefore requires a 3D perspective on the pointing act. Previous

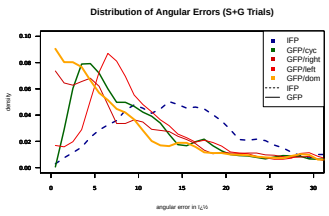
studies trying to extract these spatial information from 2D video sources failed to do so (see Section 4.1). Also, a review of the literature on manual pointing gestures revealed only qualitative descriptions. Manual pointing, for example, was identified as direction-based pointing. However, these descriptions were considered as being not satisfying, as they lacked formal rigidity and quantitative support.



To address these issues, a new formalization of the process of interpreting pointing in 3D space was introduced in Section 3.3. As a starting point, vector extrapolation and shape-based models (or better volume-based models) were formalized. The process of formalization generated important questions on the features that define the pointing direction, and on the selection and parameterization of the model describing the extension of pointing.

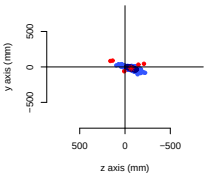


These questions were addressed in a comprehensive study on manual pointing (see Chapter 4), which generated high-precision data on pointing accuracy (see Section 4.9.2) for quantitative analysis. This was only made possible by developing new methods to collect and analyze 3D data on pointing acts (see Section 4.6 on the I*nteractive* A*ugmented* D*ata* E*xplorer* (IADE)). As a first important consequence of the found low accuracy of manual pointing, vector-extrapolation models for the extension of pointing can be discarded. Pointing direction was best described by a model that takes the direction of gaze into account (Gaze-Finger Pointing, see Section 4.9.2) and not by the alternative model based on the orientation of the index finger (Index-Finger Pointing). The study provided further results on the precision of pointing that substantiated the model of the extension of pointing. An important finding is the dichotomization of the gesture space into a proximal and a distal area (see Section 4.9.1).



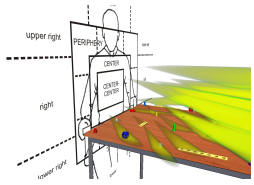
The findings of the study were further generalized in Chapter 6, where a data-driven model for pointing was developed. A change of the frame of reference improved the analysis of the data decisively (see Section 6.1). The partitioning of the gesture space into a proximal and distal area was thus further substantiated. The GFP model for the direction of pointing was elaborated into more fine-grained models (*GFP/left*, *GFP/right*, *GFP/dom*). The big break was achieved by tak-

ing into account the role of the dominant eye in pointing (see Section 6.2). The developed *GFP/dom* model provides the most accurate model for describing the direction of a manual pointing gesture.

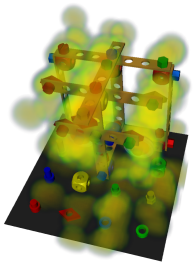


The direction of gaze was identified as a major parameter for manual pointing, but gaze is also used for deictic references on its own account. The literature coincides in associating the direction of gaze pointing with the direction of the visual axis of the eye. It was, however, shown that observing an interlocutor's gaze could also reveal the location of the point of regard. The technical applicability of this claim was verified in two studies on gaze pointing, which tested a direction-based and a location-based pointing model (see Chapter 5). The new developed algorithm for estimating the 3D point of regard of a moving observer successfully narrowed down the location to a small volume in 3D space.

Answers to the *Which* Question



Regarding manual pointing gestures, the presented findings (see Chapter 4) clearly showed that vector-extrapolation is not an appropriate model to identify which objects are the target of a pointing gesture. Based on the optimal description of the direction of manual pointing gestures by the *GFP/dom* model, new volume-based models were deviced that were found to be more accurate on the data from the study. After a thorough analysis of the data (see Chapter 6), a hybrid model was designed, which provides a very accurate description of the human capabilities in interpreting manual pointing gestures (see Section 6.3.3). This model combines the findings regarding the relevance of eye dominance for determining the direction of pointing (*GFP/dom*) with the findings about the dichotomy into proximal and distal pointing.



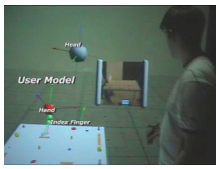
It was shown that volume-based approaches can successfully be used for selecting the targets of gaze pointing (see Section 5.9). For the first time, a machine-learning approach (PSOM) was integrated into a framework for motion tracking to compute the 3D position of the point of regard. This proved to be superior to an explicit solution based on linear algebra (see Chapter 5.6). Based on the 3D point of regard, an Attention Volume model (see Section 6.4.1) was created to compute the distribution of

attention over a certain area in 3D space. This model can be used to identify objects which received the most attention within a certain frame of time as targets of gaze pointing.

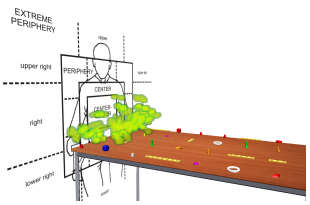
Overall, this thesis contributes a formalized approach to modeling pointing gestures and provides very accurate models for gaze and manual pointing. Based on high resolution multimodal data, established assumptions, such as "pointing as a vector", are overthrown and replaced by more adequate volume-based models.

The models presented in this thesis (see Chapter 6) are based on data from very specific domains. The objects used as targets in the studies have very distinct shapes and colors; their arrangement in space is highly artificial. The estimated quantitative parameters therefore have to be considered with care and external validations in different, more natural, settings are required. However, it is reasonable to assume that the found general principles, such as the dichotomy into proximal and distal pointing, the corresponding best measurements (angular and orthogonal), the interaction between speech and gesture or the importance of the dominant eye for determining the direction of pointing scale to other domains as well.

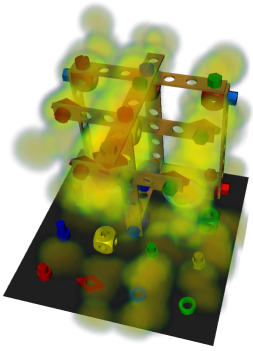
7.2.2 Contributions to Scientific Methods



The main achievement of this thesis are the detailed answers to the scientific questions on pointing. These answers consist of pointing models based on data with a precision hitherto exceptional for linguistic studies on multimodal interaction. This high level of precision was only made possible by developing novel empirical methodologies (see Chapter 4), bringing together well-tried linguistic methods with state-of-the-art tracking technology and virtual reality. The developed I*nteractive* A*ugmented* D*ata* E*xplorer* (IADE) (see Section 4.6) not only provides a sound technical basis for the recording and integration of multimodal data, it also offers exciting new possibilities to the researcher. The collected data can be augmented by manual as well as by automatic annotations, and different model hypotheses can be played out in the data-driven computer simulation. In doing so, the scientist can immerse into the data in 3D and is freed from prior restrictions imposed by the perspective of the camera once chosen. Also, by introducing virtual artifacts representing the important features that have been identified (e.g. by annotations), the objects of investigation become literally graspable.



The analysis of multimodal interaction data was supported by a newly developed visualization of the gesture space in 3D, the *Gesture Space Volumes* (GSVs, see Section 4.10.2). Gesture Space Volumes show the positions in the gesture space that have been traversed during the course of gesticulation. The GSV provides an integrated view of the trajectory of a single gesture over time, several gestures of an individual, or an aggregated view of all gestures from several interlocutors. The GSVs were used to visualize the aggregated gesture space over all trials from the study on manual pointing gestures (see Chapter 4) by which different strategies were revealed. In particular, the two primary coping strategies adopted by the participants of the study were found using GSVs: leaning-forward and raising-high.

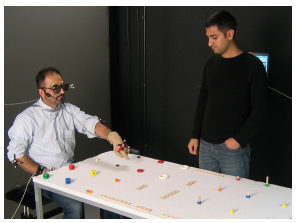


With the developed *Attention Volumes* (see Section 6.4.1, scientists can now visualize the distribution of visual attention in 3D scenes, without being restricted to stimuli which can be presented in 2D on a computer screen. Knowledge about the distribution and timecourse of visual attention (see also the 3D Scanpaths in Section 5.11.1) is crucial for many research questions in cognitive science and human-computer interaction. The Visual World paradigm (Tanenhaus, Spivey-Knowlton, Eberhard & Sedivy, 1995), for example, is a popular method used in psycholinguistic studies to investigate linguistic processes by observing the gaze path over a selection of objects constituting the visual world. The timing of the gaze path while listening to referential expressions, for example, is then used to create process models of speech understanding. Other examples include the analysis of product design by observing the distribution of attention over the object, a method frequently used in usability research. The algorithms for binocular eye tracking of a freely moving observer (see Section 5.1 and Section 5.6) thereby provide the basis for calculating the 3D point of regard during studies – either on virtual objects (see Section 5.6) or on real objects (see Section 7.1.4).

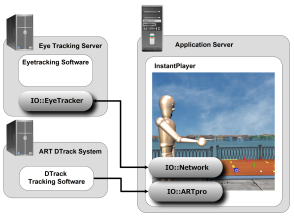
Overall, the contributed scientific methods constitute an important step towards better means to analyze human behavior in natural 3D environments. The ultimate goal is to overcome the reduction of our 3D world to 2D or 2.5D stimuli in experiments. This reduction is often enforced not by scientific demands, but by the restrictions of the technology at hand.

7.2.3 Contributions to Human-Computer Interaction

Gaze pointing is expected to be more precise and much faster than manual pointing. The DRIVE framework (see Chapter A) was created to allow for gaze interaction in a 3D environment. The studies showed (see Section 5.1 and Section 5.6) that fast response times and a high accuracy can be achieved. A unique feature of the framework is the very precise estimation of the 3D point of regard based on binocular eye tracking.



The models for the interpretation of manual pointing implemented in DRIVE are grounded in a full-scale analysis of manual pointing acts in a direct human-human interaction (see Chapter 4 and Chapter 6). This approach contrasts strongly with the approaches followed by others in this area, who either work only constructively (Fröhlich & Wachsmuth, 1998; Latoschik & Wachsmuth, 1998), eventually with post-hoc evaluations (Olwal et al., 2003), or work with data from highly restricted, non-interactive studies to ground their models on (Müller-Tomfelde, 2009).



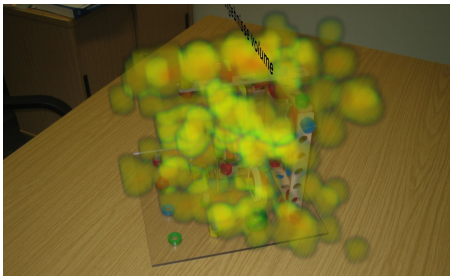
The DRIVE framework itself is based on a data-flow approach, which is a common design pattern in highly reactive interactive applications. The flexibility and the broad scope of the DRIVE framework was demonstrated in several applications: direct multimodal interaction (see Section 7.1.1), communication with Embodied Conversational Agents (see Section 6.5 and Section 7.1.2), attention-aware interfaces (see Section 7.1.3) and as data recording tool for acquiring data on 3D manual pointing (see Section 4.6) and 3D attention distributions in real world settings (see Section 7.1.4).

Overall, this thesis' focus on natural interaction with conversational interfaces gave reasons for an interdisciplinary approach to create the DRIVE framework, grounding the model design on findings from linguistics and psychology, as well as on own studies on natural communication behavior.

7.3 Further Perspectives

This thesis provides new insights on how multimodal deixis works. These insights are supported by findings based on precise measurements and simulations. They thereby complement existing work based on qualitative analysis and theoretical consideration. The findings presented in Chapter 4 and Chapter 6 have to be carefully reconsidered in the context of the scientific discourse in linguistics, as they in particular challenge the traditional doctrine of deixis as prototype for reference.

The focus of this thesis is on describing and interpreting pointing gestures. It would, however, be interesting to see, how these models perform in gesture production. This would require an implementation of the models to support the gesture planning system of an Embodied Conversational Agent, such as Max, and an evaluation in an appropriate setting. The setting used for the study presented in Chapter 4 could be used as a baseline, which would allow for a comparison between the original results for human-human interactions with a human description giver and the agent-human interactions with an artificial description giver.



In Section 7.1.4 a short example has been given, where DRIVE has been used in a context beyond conversational interfaces to track the visual attention of humans in 3D on real-world objects. This is a promising and novel approach. The advantage of the developed algorithms is that they do not require an expensive modeling of the real-world, but can work out-of-the-box on any objects or scenes. However, the set-up and calibration of the system needs to be improved to be more robust and support larger areas of investigation. Up to now, the tracking is done outside-in using a tracking system based on optical markers. For real-world settings, inside-out tracking systems, such as those being used for augmented reality, could increase flexibility and reduce overall costs. A robust tracking of visual attention in space would be beneficial for basic research in psychology, biology and linguistics, as well as for the design and usability evaluation of human-computer interactions, especially in the areas of ambient intelligence and robotics.

Overall, the models for pointing developed in this thesis exceed other existing model in terms of technical formality, precision and accuracy. The invented scientific methods provide new technical means for studies on human commu-

nication and visual attention in our 3D world. The technical soundness of the DRIVE framework has been proven in several applications. Finally, the models implemented in DRIVE have an exceptional level of empirical validity. DRIVE thus provides a solid foundation for more natural conversational interfaces.

Bibliography

Advanced Realtime Tracking GmbH (2010). DTrack. Retrieved April 2010 from `http://www.ar-tracking.de`. [46, 72, 105]

Anliker, J. (1976). Eye movements: On-Line Measurement, Analysis, and Control. In R. A. Monty & J. W. Senders (Eds.), *Eye Movements and Psychological Processes* (pp. 185–202). Hillsdale, NJ: Lawrence Erlbaum Associates. [51]

Arrington Research Inc. (2008). Arrington Research. Retrieved April 2010 from `http://www.arringtonresearch.com/`. [105]

Banks, M., Ghose, T., & Hillis, J. (2004). Relative image size, not eye position, determines eye dominance switches. *Vision Research*, *44*, 229–234. [138]

Barabas, J., Goldstein, R., Apfelbaum, H., Woods, R., Giorgi, R., & Peli, E. (2004). Tracking the line of primary gaze in a walking simulator: modeling and calibration. *Behavior Research Methods, Instruments, & Computers*, *36(4)*, 757–770. [55]

Barakonyi, I., Prendinger, H., Schmalstieg, D., & Ishizuka, M. (2007). Cascading Hand and Eye Movement for Augmented Reality Videoconferencing. In *Proc. 2nd IEEE Symposium on 3D User Interfaces 2007 (3DUI 2007)*, (pp. 71–78)., Charlotte, North Carolina, USA. [56]

Bühler, K. (1934). *Sprachtheorie: Die Darstellungsform der Sprache*. Jena: Gustav Fischer. [13, 16]

Biermann, P., Jung, B., Latoschik, M., & Wachsmuth, I. (2002). Virtuelle Werkstatt: A Platform for Multimodal Assembly in VR. In *Proceedings Fourth Virtual Reality International Conference (VRIC 2002),Laval, France, 19-21 June 2002*, (pp. 53–62). [76, 217]

Bluhm, A., Eickmeyer, J., Feith, T., Mattar, N., & Pfeiffer, T. (2009). Exploration von sozialen Netzwerken im 3D Raum am Beispiel von SoN∀R für Last.fm. In Gerndt, A. & Latoschik, M. E. (Eds.), *Virtuelle und*

Erweiterte Realität - Sechster Workshop der GI-Fachgruppe VR/AR, (pp. 269–280)., Aachen. Shaker Verlag. [171]

Boersma, P. (2001). Praat, a system for doing phonetics by computer. *Glot international*, *5*(9/10), 341–345. [78]

Bojko, A. (2009). Informative or Misleading? Heatmaps Deconstructed. In Jacko, J. A. (Ed.), *Human-Computer Interaction. New Trends, 13th International Conference, HCI International 2009, San Diego, CA, USA, July 19-24, 2009, Proceedings, Part I*, volume 5610 of *Lecture Notes in Computer Science*, (pp. 30–39). Springer. [30]

Bolt, R. (1980). Put-That-There: Voice and gesture at the graphics interface. In *ACM SIGGRAPH - Computer Graphics*, (pp. 262–270)., New York. ACM Press. [36, 37]

Bolt, R. (1981). Gaze-orchestrated dynamic windows. *Proceedings of the 8th annual conference on Computer graphics and interactive techniques*, 109–119. [36, 37]

Bolt, R. (1982). Eyes at the interface. *Proceedings of the 1982 conference on Human factors in computing systems*, 360–362. [36, 37]

Bowman, D. A., Kruijff, E., Joseph J. LaViola, J., & Poupyrev, I. (2005). *3D User Interfaces – Theory and Practice*. Addison-Wesley. [35]

Butterworth, G. (2003). Pointing is the royal road to language for babies. In S. Kita (Ed.), *Pointing: Where Language, Culture, and Cognition Meet* chapter 2, (pp. 9–33). Mahwah, New Jersey: Lawrence Erlbaum Associates. [13, 14, 22]

Butterworth, G. & Itakura, S. (2000). How the eyes, head and hand serve definite reference. *British Journal of Developmental Psychology*, *18*, 25–50. [20, 21, 22, 24, 32]

Card, S., Moran, T., & Newell, A. (1983). *The Psychology of Human-Computer Interaction*. Lawrence Erlbaum Associates. [113]

Cassell, J., Bickmore, T., Billinghurst, M., Campbell, L., Chang, K., Vilhjálmsson, H., & Yan, H. (1998). An architecture for embodied conversational characters. In *Proceedings of the First Workshop on Embodied Conversational Characters, Tahoe City, California October 1998*, (pp. 109–120). [43]

Cassell, J., Sullivan, J., Prevost, S., & Churchill, E. (Eds.). (2000). *Embodied Conversational Agents*. Cambridge, MA: MIT Press. [39]

Chaurasia, B. & Mathur, B. (1976). Eyedness. *Cells Tissues Organs*, *96*(2), 301–305. [139]

Chi, C.-F. & Lin, C.-L. (1997). Aiming accuracy of the line of gaze and redesign of the gaze-pointing system. *Perceptual and motor skills*, *85*(3 Pt 1), 1111. [25, 111]

Cleveland, W. (1981). LOWESS: A program for smoothing scatterplots by robust locally weighted regression. *American Statistician*, *35*(1), 54. [141]

Codella, C., Jalili, R., Koved, L., Lewis, J. B., Ling, D. T., Lipscomb, J. S., Rabenhorst, D. A., Wang, C. P., Norton, A., Sweeney, P., & Turk, G. (1992). Interactive simulation in a multi-person virtual world. In *CHI '92: Proceedings of the SIGCHI conference on Human factors in computing systems*, (pp. 329–334)., New York, NY, USA. ACM. [55]

Conover, W. J. (1971). *Practical nonparametric statistics*. New York: John Wiley & Sons. [119]

Corazza, S., Mündermann, L., Chaudhari, A., Demattio, T., Cobelli, C., & Andriacchi, T. (2006). A markerless motion capture system to study musculoskeletal biomechanics: Visual hull and simulated annealing approach. *Annals of Biomedical Engineering*, *34*(6), 1019–1029. [46]

Cournia, N., Smith, J. D., & Duchowski, A. T. (2003). Gaze- vs. hand-based pointing in virtual environments. In *Conference on Human Factors in Computing Systems, CHI '03*, (pp. 772 – 773)., New York, NY. ACM Press. [52]

CyberGlove Systems (1990). CyberGlove. Retrieved April 2010 from `http://www.cyberglovesystems.com/`. [46, 72]

Dähne, P. (2009). InstantIO Documentation. Retrieved April 2010 from `http://www.instantreality.org/device/`. [203]

Duchowski, A. T. (2007). *Eye Tracking Methodology: Theory and Practice* (Second Edition ed.). Springer-Verlag London Limited. [24, 25, 51]

Duchowski, A. T., Medlin, E., Cournia, N., Murphy, H., Gramopadhye, A., Nair, S., Vorah, J., & Melloy, B. (2002). 3D Eye Movement Analysis. *Behavior Research Methods, Instruments and Computers*, *34*(4), 573–591. [54, 56, 107, 173]

Eichner, T., Prendinger, H., André, E., & Ishizuka, M. (2007). Attentive Presentation Agents. In Pelachaud, C., Martin, J.-C., André, E., Chollet,

G., Karpouzis, K., & Pelé, D. (Eds.), *IVA '07: Proceedings of the 7th international conference on Intelligent Virtual Agents*, (pp. 283–295)., Berlin, Heidelberg. Springer-Verlag. [44]

Essig, K., Pomplun, M., & Ritter, H. (2006). A neural network for 3D gaze recording with binocular eye trackers. *The International Journal of Parallel, Emergent and Distributed Systems*, *21*(2), 79–95. [58, 60, 118, 122, 123]

Fellner, D., Behr, J., & Bockholt, U. (2009). Instantreality - a framework for industrial augmented and virtual reality applications. In *The 2nd Sino-German Workshop "Virtual Reality & Augmented Reality in Industry" : Invited Paper Proceedings. Participants Edition.*, (pp. 78–83)., Shanghai Jiao Tong University. [162, 201]

Fillmore, C. J. (1975). *Santa Cruz Lectures on Deixis 1971.* University of California, Berkeley: Indiana University Linguistics Club. [12]

Flitter, H., Pfeiffer, T., & Rickheit, G. (2006). Psycholinguistic experiments on spatial relations using stereoscopic presentation. In G. Rickheit & I. Wachsmuth (Eds.), *Situated Communication* (pp. 127–153). Berlin: Mouton de Gruyter. [116]

Fröhlich, M. & Wachsmuth, I. (1998). Gesture recognition of the upper limbs - from signal to symbol. In I. Wachsmuth & M. Fröhlich (Eds.), *Gesture and Sign Language in Human-Computer Interaction*, LNAI 1371 (pp. 173–184). Springer-Verlag. [37, 180]

Gale, C. & Monk, A. (2000). Where am I looking? The accuracy of video-mediated gaze awareness. *Perception and Psychophysics*, *62*(3), 586–595. [28]

Goldstein, E. B. (2002). *Wahrnehmungspsychologie.* Spektrum Akademischer Verlag. [26]

Goodwin, C. (2000). Gesture, aphasia, and interaction. In D. McNeill (Ed.), *Language and gesture* (pp. 84–98). Cambridge University Press. [16]

Gullberg, M. & Holmqvist, K. (1999). Keeping an eye on gestures: Visual perception of gestures in face-to-face communication. *Pragmatics & Cognition*, *7*(1), 35–63. [41]

Gullberg, M. & Holmqvist, K. (2002). *Gesture and Sign Language in Human-Computer Interaction*, chapter Visual attention towards gestures in face-to-face interaction vs. on screen, (pp. 206–214). LNAI. Berlin Heidelberg: Springer. [41]

Haviland, J. B. (2000). Pointing, gesture spaces, and mental maps. In D. McNeill (Ed.), *Language and gesture* (pp.1̃3). Cambridge: Cambridge University Press. [15, 16]

Herrmann, T. (1982). *Sprechen und Situation: Eine psychologische Konzeption zur situationsspezifischen Sprachproduktion*. Springer. [10]

Hillaire, S., Lecuyer, A., Cozot, R., & Casiez, G. (2008). Using an Eye-Tracking System to Improve Camera Motions and Depth-of-Field Blur Effects in Virtual Environments. *Virtual Reality Conference, 2008. VR'08. IEEE*, 47–50. [39]

Hollander, M. & Wolfe, D. A. (1973). *Nonparametric statistical inference*. New York: John Wiley & Sons. [119]

Humphreys, G., Houston, M., Ng, R., Frank, R., Ahern, S., Kirchner, P. D., & Klosowski, J. T. (2002). Chromium: a stream-processing framework for interactive rendering on clusters. *ACM Trans. Graph.*, *21*(3), 693–702. [105]

Hutchins, E. (1987). Metaphors for Interface Design. Ics report 8703, Insitute for Cognitive Science, University of California, San Diego. [39]

Hutchinson, T., White Jr, K., Martin, W., Reichert, K., & Frey, L. (1989). Human-computer interaction using eye-gaze input. *IEEE Transactions on systems, man and cybernetics*, *19*(6), 1527–1534. [38]

ISO 19775-1:2004 (2004). *Part 1: Architecture and base components: Information technology – Computer graphics and image processing – Extensible 3D (X3D)*. ISO, Geneva, Switzerland. [125, 162, 201]

Jacob, R. J. K. (1993). *Eye-movement-based human-computer interaction techniques: Toward non-command interfaces*, volume 4 of *Advances in Human-Computer Interaction*, chapter 6, (pp. 151–190). Norwood, New Jersey: Ablex Publishing Corporation. [38]

Jacob, R. J. K. & Karn, K. S. (2003). Eye Tracking in Human-Computer Interaction and Usability Research: Ready to Deliver the Promises (Section Commentary). In J. Hyona, R. Radach, & H. Deubel (Eds.), *The Mind's Eye: Cognitive and Applied Aspects of Eye Movement Research* (pp. 573–605). Amsterdam: Elsevier Science. [38]

Jung, B., Hoffhenke, M., & Wachsmuth, I. (1998). Virtual Assembly with Construction Kits. In *Proceedings of the 1998 ASME Design for Engineering Technical Conferences (DECT-DFM '98)*, Sacramento, CA. Assembly with

Construction Kits. In Proceedings of the 1998 ASME Design for Engineering arTechnical Conferences (DECT-DFM '98), 1998. [167]

Kaiser, E., Olwal, A., McGee, D., Benko, H., Corradini, A., Li, X., Cohen, P., & Feiner, S. (2003a). Mutual disambiguation of 3D multimodal interaction in augmented and virtual reality. In *ICMI '03: Proceedings of the 5th international conference on Multimodal interfaces*, (pp. 12–19)., New York, NY, USA. ACM Press. [55, 56]

Kaiser, E., Olwal, A., McGee, D., Benko, H., Corradini, A., Li, X., Cohen, P., & Feiner, S. (2003b). Mutual Disambiguation of 3D Multimodal Interaction in Augmented and Virtual Reality. In *Proceedings of the 5th International Conference on Multimodal Interfaces, ICMI 2003, Vancouver, British Columbia, Canada, November 5-7*, (pp. 12–19). ACM Press. [61]

Kendon, A. (1980). Gesticulation and Speech: Two Aspects of the Process of Utterance. In M. R. Key (Ed.), *The relation between verbal and non-verbal communication* (pp. 207–227). The Hague: Mouton. [15, 19]

Kendon, A. (2004). *Gesture: Visible action as utterance*. Cambridge, UK: Cambridge University Press. [14, 16]

Kipp, M. (2001). Anvil - a generic annotation tool for multimodal dialogue. In *Seventh European Conference on Speech Communication and Technology*. [online] `http://www.anvil-software.de`. [78]

Kita, S. (1990). The temporal relationship between gesture and speech: A study of Japanese-English bilinguals. Master's thesis, Department of Psychology, University of Chicago. [15]

Kita, S. (2003). *Pointing: Where language, culture, and cognition meet*. Mahwah, New Jersey: L. Erlbaum Associates. [16]

Kohonen, T. (1990). The self-organizing map. *Proceedings of IEEE*, *78*(9), 1464–1480. [59]

Koons, D., Sparrel, C., & Thorisson, K. (1993). *Integrating simultaneous input from speech, gaze and hand gestures*. AAAI Press. [60]

Kopp, S. (2003). *Synthese und Koordination von Sprache und Gestik fuer Virtuelle Multimodale Agenten*. Infix DISKI-265. Berlin: Akademische Verlagsgesellschaft Aka GmbH. [41]

Kopp, S., Sowa, T., & Wachsmuth, I. (2004). *Gesture-Based Communication in Human-Computer Interaction*, volume 2915/2004 of *LNAI*, chapter Imitation games with an artificial agent: From mimicking to understanding

shape-related iconic gestures, (pp. 436–447). Berlin Heidelberg: Springer. [42, 44]

Kopp, S., Tepper, P., & Cassell, J. (2004). Towards Integrated Microplanning of Language and Iconic Gesture for Multimodal Output. In *International Conference on Multimodal Interfaces (ICMI)*, (pp. 97–104). ACM Press. [42]

Kranstedt, A. (2007). *Situierte Generierung deiktischer Objektreferenz in der multimodalen Mensch-Maschine-Interaktion*. DISKI 313. Berlin: Akademische Verlagsgesellschaft Aka GmbH. [41, 47, 80, 148]

Kranstedt, A., Kopp, S., & Wachsmuth, I. (2002). MURML: A Multimodal Utterance Representation Markup Language for Conversational Agents. In *Proceedings of the Workshop Embodied conversational agents - let's specify and evaluate them!, held at the First Int. Joint Conference on Autonomous Agents & Multi-Agent Systems*, Bologna, Italy. [41]

Kranstedt, A., Kühnlein, P., & Wachsmuth, I. (2004). Deixis in Multimodal Human Computer Interaction: An Interdisciplinary Approach. In Camurri, A. & Volpe, G. (Eds.), *Gesture-Based Communication in Human-Computer Interaction, International Gesture Workshop, Genua, Italy April 2003*, LNAI 2915, (pp. 112–123). Springer. [66]

Kranstedt, A., Lücking, A., Pfeiffer, T., Rieser, H., & Staudacher, M. (2006). Measuring and Reconstructing Pointing in Visual Contexts. In Schlangen, D. & Fernández, R. (Eds.), *Proceedings of the brandial 2006 - The 10th Workshop on the Semantics and Pragmatics of Dialogue*, (pp. 82–89)., Potsdam. Universitätsverlag Potsdam. [47, 78]

Kranstedt, A., Lücking, A., Pfeiffer, T., Rieser, H., & Wachsmuth, I. (2006a). Deictic object reference in task-oriented dialogue. In G. Rickheit & I. Wachsmuth (Eds.), *Situated Communication* (pp. 155–207). Berlin: Mouton de Gruyter. [42, 80]

Kranstedt, A., Lücking, A., Pfeiffer, T., Rieser, H., & Wachsmuth, I. (2006b). Deixis: How to Determine Demonstrated Objects Using a Pointing Cone. In Gibet, S., Courty, N., & Kamp, J.-F. (Eds.), *Gesture Workshop 2005*, LNAI 3881, (pp. 300–311)., Berlin Heidelberg. Springer-Verlag GmbH. [80]

Kühnlein, P. & Stegmann, J. (2003). Empirical issues in deictic gesture: referring to objects in simple identification tasks. *Report 2003/3, SFB, 360*. [80]

Kwon, Y.-M., Jeon, K.-W., Ki, J., Shahab, Q. M., Jo, S., & Kim, S.-K. (2006). 3D Gaze Estimation and Interaction to Stereo Display. *The International Journal of Virtual Reality*, *5*(3), 41–45. [56]

Lanier, J. & Zimmermann, T. (1986). DataGlove. company insolvent in 1993. [46]

Latoschik, M. (2002). Designing Transition Networks for Multimodal VR-Interactions Using a Markup Language. In *Proceedings of the IEEE fourth International Conference on Multimodal Interfaces, ICMI 2002, Pittsburgh, USA, October 2002*, (pp. 411–416). [37, 60, 61]

Latoschik, M. E. (2001). A General Framework for Multimodal Interaction in Virtual Reality Systems: PrOSA. In Broll, W. & Schäfer, L. (Eds.), *The Future of VR and AR Interfaces - Multimodal, Humanoid, Adaptive and Intelligent. Proceedings of the workshop at IEEE Virtual Reality 2001, Yokohama, Japan*, (pp. 21–25)., Sankt Augustin. GMD Report No. 138, GMD-Forschungszentrum Informationstechnik GmbH. [37]

Latoschik, M. E., Fröhlich, M., Jung, B., & Wachsmuth, I. (1998). Utilize Speech and Gestures to Realize Natural Interaction in a Virtual Environment. In *IECON'98 - Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society*, volume 4, (pp. 2028–2033). IEEE. [37]

Latoschik, M. E. & Wachsmuth, I. (1998). Exploiting distant pointing gestures for object selection in a virtual environment. In I. Wachsmuth & M. Fröhlich (Eds.), *Gesture and Sign Language in Human-Computer Interaction*, LNAI 1371 (pp. 185–196). Springer-Verlag. [37, 47, 48, 180]

Lee, J., Marsella, S., Traum, D. R., Gratch, J., & Lance, B. (2007). The Rickel Gaze Model: A Window on the Mind of a Virtual Human. In Pelachaud, Martin, André, Chollet, Karpouzis & Pelé (2007), (pp. 296–303). [40]

Lee, S. P., Badler, J. B., & Badler, N. I. (2002). Eyes alive. In *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, (pp. 637–644)., New York, NY, USA. ACM Press. [40]

Lessmann, N., Kranstedt, A., & Wachsmuth, I. (2004). Towards a Cognitively Motivated Processing of Turn-Taking Signals for the Embodied Conversational Agent Max. In *AAMAS 2004 Workshop Proceedings: "Embodied Conversational Agents: Balanced Perception and Action"*, (pp. 57–64)., New York. [43]

Levelt, W., Richardson, G., & La Heij, W. (1985). Pointing and Voicing in Deictic Expressions. *Journal of Memory and Language*, *24*(2), 133–164. [19, 20]

Levine, J. (1981). An eye-controlled computer. *Research Report RC-8857, IBM Thomas J. Watson Research Center, Yorktown Heights, NY*. [38]

Levine, J. (1984). Performance of an eyetracker for office use. *Computers in biology and medicine*, *14*(1), 77. [38]

Lewis, J. B., Koved, L., & Ling, D. T. (1991). Dialogue structures for virtual worlds. In *CHI '91: Proceedings of the SIGCHI conference on Human factors in computing systems*, (pp. 131–136)., New York, NY, USA. ACM. [55]

Lücking, A., Rieser, H., & Stegmann, J. (2004). Statistical support for the study of structures in multimodal dialogue: Inter-rater agreement and synchronization. In *Proceedings of the 8th Workshop on the Semantics and Pragmatics of Dialogue*, (pp. 56–63). [66]

Luebke, D., Hallen, B., Newfield, D., & Watson, B. (2000). Perceptually Driven Simplification Using Gaze-Directed Rendering. Technical report, University of Virginia Technical Report, University of Virginia. [39]

Lyons, J. (1968). *Introduction to theoretical linguistics*. Cambridge University Press. [11]

Lyons, J. (1977). *Semantics*, volume 2, chapter Deixis, space and time, (pp. 636–724). Cambridge Univ. Press. [12]

Mayberry, R. & Jaques, J. (2000). Gesture production during stuttered speech: Insights into the nature of gesture-speech integration. In D. McNeill (Ed.), *Language and gesture* (pp. 199–214). Cambridge University Press. [19]

McNeill, D. (1985). So you think gestures are nonverbal. *Psychological review*, *92*(3), 350–371. [19]

McNeill, D. (1992). *Hand and Mind: What Gestures Reveal about Thought*. Chicago: University of Chicago Press. [14, 15, 16, 19, 23, 78, 89]

Mitsugami, I., Ukita, N., & Kidode, M. (2003). Estimation of 3D Gazed Position Using View Lines. *Image Analysis and Processing, International Conference on*, *0*, 466. [58, 173]

Monk, A. & Gale, C. (2002). A Look Is Worth a Thousand Words: Full Gaze Awareness in Video-Mediated Conversation. *Discourse Processes*, *33*(3), 257–278. [28]

Morency, L.-P., Christoudias, C. M., & Darrell, T. (2006). Recognizing gaze aversion gestures in embodied conversational discourse. In *ICMI '06: Proceedings of the 8th international conference on Multimodal interfaces*, (pp. 287–294)., New York, NY, USA. ACM Press. [43]

Morrel-Samuels, P. & Krauss, R. (1992). Word familiarity predicts temporal asynchrony of hand gestures and speech. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *18*(3), 615–622. [19]

Müller-Tomfelde, C. (2009). Dwell-Based Pointing in Applications of Human Computer Interaction. In *Human-Computer Interaction – INTERACT 2007 (LNCS 4662)*, (pp. 560–573). Springer Berlin. [19, 180]

Murphy, H. & Duchowski, Andrew, T. (2001). Gaze-Contingent Level Of Detail Rendering. In *EuroGraphics*. EuroGraphics Association. [39]

Nakano, Y. I., Reinstein, G., Stocky, T., & Cassell, J. (2003). Towards a model of face-to-face grounding. *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, 553–561. [43]

NaturalPoint, Inc. (1997). OptiTrack. Retrieved April 2010 from `http://www.naturalpoint.com/optitrack`. [46]

Neggers, S. & Bekkering, H. (2000). Ocular Gaze is Anchored to the Target of an Ongoing Pointing Movement. *Journal of Neurophysiology*, *83*(2), 639–651. [30]

Neggers, S. & Bekkering, H. (2001). Gaze anchoring to a pointing target is present during the entire pointing movement and is driven by a non-visual signal. *Journal of Neurophysiology*, *86*(2), 961. [31]

Nielsen, J. (1993). Noncommand user interfaces. *Commun. ACM*, *36*(4), 83–99. [38, 39]

Nobe, S. (2000). Where do most spontaneous representational gestures actually occur with respect to speech. In D. McNeill (Ed.), *Language and gesture* (pp. 186–198). Cambridge University Press. [19]

Nobe, S., Hayamizu, S., Hasegawa, O., & Takahashi, H. (2000). Hand Gestures of an Anthropomorphic Agent: Listeners' Eye Fixation and Comprehension. *Cognitive Studies*, *7*(1), 86–92. [41]

Oh, A., Fox, H., Kleek, M. V., Adler, A., Gajos, K., Morency, L.-P., & Darrell, T. (2002). Evaluating look-to-talk: a gaze-aware interface in a collaborative environment. In *CHI '02: CHI '02 extended abstracts on Human factors in computing systems*, (pp. 650–651)., New York, NY, USA. ACM Press. [43]

Olwal, A., Benko, H., & Feiner, S. (2003). SenseShapes: Using Statistical Geometry for Object Selection in a Multimodal Augmented Reality System. In *Proceedings of The Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2003)*, (pp. 300–301)., Tokyo, Japan. [55, 180]

Organic Motion, Inc. (2007). Stage. Retrieved April 2010 from `http://www.organicmotion.com`. [46]

Papenmeier, F. & Huff, M. (2010). DynAOI: A tool for matching eye-movement data with dynamic areas of interest in animations and movies. *Behavior Research Methods*, *42*(1), 179. [173]

Papert, S. & Minsky, M. (1967). Computer Tracking of Eye Motions. Artificial Intelligence Memo. No. 123. Vision Memo. [38]

Peirce, C. S. (1965). *Collected Papers of Charles Sanders Peirce*, volume II. Cambridge, MA: Harvard University Press. repr. from 1932. [2, 3]

Pelachaud, C., Martin, J.-C., André, E., Chollet, G., Karpouzis, K., & Pelé, D. (Eds.). (2007). *Intelligent Virtual Agents, 7th International Conference, IVA 2007, Paris, France, September 17-19, 2007, Proceedings*, volume 4722 of *Lecture Notes in Computer Science*. Springer. [190, 194]

Pfeiffer, T. (2003). Eine Referenzauflösung für die dynamische Anwendung in Konstruktionssituationen in der Virtuellen Realität. Master's thesis, Faculty of Technology, University of Bielefeld. [163, 168]

Pfeiffer, T. (2008). Towards Gaze Interaction in Immersive Virtual Reality: Evaluation of a Monocular Eye Tracking Set-Up. In Schumann, M. & Kuhlen, T. (Eds.), *Virtuelle und Erweiterte Realität - Fünfter Workshop der GI-Fachgruppe VR/AR*, (pp. 81–92)., Aachen. Shaker Verlag GmbH. [104]

Pfeiffer, T., Donner, M., Latoschik, M. E., & Wachsmuth, I. (2007a). 3D fixations in real and virtual scenarios. *Journal of Eye Movement Research, Special issue: Abstracts of the ECEM 2007*, 13. [114]

Pfeiffer, T., Donner, M., Latoschik, M. E., & Wachsmuth, I. (2007b). Blickfixationstiefe in stereoskopischen VR-Umgebungen: Eine vergleichende Studie. In Latoschik, M. E. & Fröhlich, B. (Eds.), *Vierter Workshop Virtuelle und Erweiterte Realität der GI-Fachgruppe VR/AR*, (pp. 113–124)., Aachen. Shaker. [114]

Pfeiffer, T., Kranstedt, A., & Lücking, A. (2006). Sprach-Gestik Experimente mit IADE, dem Interactive Augmented Data Explorer. In Müller, S. &

Zachmann, G. (Eds.), *Dritter Workshop Virtuelle und Erweiterte Realität der GI-Fachgruppe VR/AR*, (pp. 61–72)., Aachen. Shaker. [67, 74]

Pfeiffer, T. & Latoschik, M. E. (2004). Resolving Object References in Multimodal Dialogues for Immersive Virtual Environments. In Ikei, Y., Göbel, M., & Chen, J. (Eds.), *Proceedings of the IEEE Virtual Reality 2004*, (pp. 35–42). IEEE. [61, 163, 168]

Pfeiffer, T. & Latoschik, M. E. (2007). Interactive Social Displays. In Fröhlich, B., Blach, R., & van Liere, R. (Eds.), *IPT-EGVE 2007, Virtual Environments 2007, Short Papers and Posters*, (pp. 41–42). Eurographics Association. [76]

Pfeiffer, T., Latoschik, M. E., & Wachsmuth, I. (2009). Evaluation of Binocular Eye Trackers and Algorithms for 3D Gaze Interaction in Virtual Reality Environments. *Journal of Virtual Reality and Broadcasting*, *5*(16). [114]

Pfeiffer, T., Voss, I., & Latoschik, M. E. (2003). Resolution of Multimodal Object References using Conceptual Short Term Memory. In Schmalhofer, F. & Young, R. (Eds.), *Proceedings of the EuroCogSci03*, (pp. 426). Lawrence Erlbaum Associates Inc. [168]

Pfeiffer-Lessmann, N. & Wachsmuth, I. (2008). Toward alignment with a virtual human – achieving joint attention. In Dengel, A. R., Berns, K., & Breuel, T. (Eds.), *KI 2008: Advances in Artificial Intelligence*, LNAI 5243, (pp. 292–299)., Berlin. Springer. [43, 44, 163, 169, 223]

PhaseSpace (1994). Impulse. Retrieved April 2010 from `http://www.phasespace.com`. [46]

Picot, A., Bailly, G., Elisei, F., & Raidt, S. (2007). Scrutinizing Natural Scenes: Controlling the Gaze of an Embodied Conversational Agent. In Pelachaud et al. (2007), (pp. 272–282). [40]

Polhemus, B. (1994). FASTRAK. Retrieved April 2010 from `http://www.polhemus.com`. [46]

Pomplun, M., Prestin, E., & Rieser, H. (1998). Eye-Movement Research and Dialogue Structure. Technical Report 98/12, SFB 360 - Universität Bielefeld. [38]

Qvarfordt, P. & Zhai, S. (2005). Conversing with the user based on eye-gaze patterns. In *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*, (pp. 221–230)., New York, NY, USA. ACM Press. [43]

Raidt, S., Bailly, G., & Elisei, F. (2007). Analyzing and modeling gaze during face-to-face interaction. In *Proceedings of the International Conference on Auditory-Visual Speech Processing (AVSP'07)*. [40]

Raidt, S., Elisei, F., & Bailly, G. (2005). Face-to-face interaction with a conversational agent: eye-gaze and deixis. In *International Conference on Autonomous Agents and Multiagent Systems*, The Netherlands. Utrecht University. [40]

Rickheit, G. & Wachsmuth, I. (1996). Collaborative Research Centre "Situated Artificial Communicators" at the University of Bielefeld, Germany. *Artificial Intelligence Review, 10 (3-4)*, 165–170. [4]

Rickheit, G. & Wachsmuth, I. (2008). Alignment in communication – Collaborative Research Center 673 at Bielefeld University. *Künstliche Intelligenz, Heft 2/2008*, 62–65. [4]

Ritter, H. (1993). Parametrized self-organizing maps. *Proceedings of ICANN93*, 568–577. [58]

Rötting, M., Göbel, M., & Springer, J. (1999). Automatic object identification and analysis of eye movement recordings. *MMI-Interaktiv, 2*. [38]

Rousseeuw, P., Ruts, I., & Tukey, J. (1999). The Bagplot: A Bivariate Boxplot. *American Statistician, 53*(4), 382–387. [82, 119]

Salvucci, D. & Goldberg, J. (2000). Identifying fixations and saccades in eye-tracking protocols. In *Proceedings of the 2000 symposium on Eye tracking research & applications*, (pp. 71–78). ACM New York, NY, USA. [207, 208]

Scully, J. & Blood, E. (1986). Ascension Technology Corporation. Retrieved April 2010 from `http://www.ascension-tech.com/`. [46]

Sennholz, K. (1985). Grundzüge der Deixis. In *Bochumer Beiträge zur Semiotik*. Bochum: Studienverlag Dr. Norbert Brockmeyer. [82]

Shannon, C. E. (1948). A Mathematical Theory of Communication. *The Bell System Technical Journal, 27*, 379–423, 623–656. [2, 11]

Shneiderman, B. (1982). The future of interactive systems and the emergence of direct manipulation. *Behaviour & Information Technology, 1*(3), 237–256. [39]

Sowa, T. (2006). *Understanding Coverbal Ionic Gestures in Shape Descriptions*. Infix DISKI-294. Berlin: Akademische Verlagssgesellschaft Aka GmbH. [44]

Sowa, T., Fröhlich, M., & Latoschik, M. (1999). Temporal Symbolic Integration Applied to a Multimodal System Using Gestures and Speech. In Braffort, A. et al. (Eds.), *Gesture-Based Communication in Human-Computer Interaction -Proceedings International Gesture Workshop (Gif-sur-Yvette, France, March 1999)*, LNAI 1739, (pp. 291–302). Springer-Verlag. [37]

Sparrell, C. J. & Koons, D. B. (1994). Interpretation of coverbal depictive gestures. In *AAAI Spring Symposium Series*, (pp. 8–12)., Stanford University. [60]

Starker, I. & Bolt, R. A. (1990). A gaze-responsive self-disclosing display. In *CHI '90: Proceedings of the SIGCHI conference on Human factors in computing systems*, (pp. 3–10)., New York, NY, USA. ACM. [37]

Suryakumar, R., Meyers, J. P., Irving, E. L., & Bobier, W. R. (2007). Application of video-based technology for the simultaneous measurement of accommodation and vergence. *Vision research(Oxford)*, *47*(2), 260–268. [27]

Tanenhaus, M. K., Spivey-Knowlton, M. J., Eberhard, K. M., & Sedivy, J. C. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science*, *268*, 1632–1634. [179]

Tanriverdi, V. & Jacob, R. J. K. (2000). Interacting with eye movements in virtual environments. In *Conference on Human Factors in Computing Systems, CHI 2000*, (pp. 265–272)., New York. ACM Press. [52]

Thórisson, K. R. (1997). Gandalf: an embodied humanoid capable of real-time multimodal dialogue with people. In *AGENTS '97: Proceedings of the first international conference on Autonomous agents*, (pp. 536–537)., New York, NY, USA. ACM. [43]

Torres, O., Cassell, J., & Prevost, S. (1997). Modeling Gaze Behavior as a Function of Discourse Structure. *Paper presented at the First International Workshop on Human-Computer Conversation.* [40]

Tramberend, H. (2001). Avango: A Distributed Virtual Reality Framework. In *Proceedings of Afrigraph '01.* ACM. [105]

Traum, D. & Rickel, J. (2002). Embodied agents for multi-party dialogue in immersive virtual worlds. In *Proceedings First Int. Joint Conference on Autonomous Agents and Multiagent systems*, (pp. 766–773). [43]

Triesch, J., Brian T. Sullivan, B. T., Hayhoe, M. M., & Ballard, D. H. (2002). Saccade contingent updating in virtual reality. In *Eye Tracking Research*

& Application: Proceedings of the symposium on Eye tracking research & applications, (pp. 95 – 102)., New Orleans, Louisiana. ACM Press. [39]

Velichkovsky, B., Sprenger, A., & Pomplun, M. (1997). Auf dem Weg zur Blickmaus: Die Beeinflussung der Fixationsdauer durch kognitive und kommunikative Aufgaben. *Software-Ergonomie. Stuttgart: Teubner.* [27, 28, 48, 51, 210, 211]

Vertegaal, R. P. H., Slagter, R., van der Veer, G. C., & Nijholt, A. (2001). Eye Gaze Patterns in Conversations: There is More to Conversational Agents Than Meets the Eyes. In Jacko, J., Sears, A., Beaudouin-Lafon, M., & Jacob, R. J. K. (Eds.), *Proceedings ACM SIGCHI Conference CHI 2001: Anyone. Anywhere, Seattle, USA*, (pp. 301–308)., New York. ACM Press. [40, 42]

Vicon Motion Systems (1984). Homepage. Retrieved April 2010 from `http://www.vicon.com`. [46]

Wachsmuth, I., Lenzmann, B., Jörding, T., Jung, B., Latoschik, M., & Fröhlich, M. (1997). A virtual interface agent and its agency. *Proceedings of the First International Conference on Autonomous Agents*, 516–517. [37]

Wheatstone, C. (1838). Contributions to the Physiology of Vision. – Part the First. On some remarkable, and hitherto unobserved, Phenomena of Binocular Vision. *Philosophical Transactionsof the Royal Society of London, 128*, 371–394. [27]

Wingrave, C. & Bowman, D. (2005). Baseline Factors for Raycasting Selection. *Proceedings of Virtual Reality International.* [55]

Wingrave, C. A., Bowman, D. A., & Ramakrishnan, N. (2002). Towards preferences in virtual environment interfaces. In *EGVE '02: Proceedings of the workshop on Virtual environments 2002*, (pp. 63–72)., Aire-la-Ville, Switzerland. Eurographics Association. [48]

Xsens Technologies B.V. (2009). Homepage. Retrieved April 2010 from `http://www.xsens.com`. [45]

Appendix

DRIVE: Deictic Reference in Virtual Environments

This chapter demonstrates how the model for gaze and gesture pointing developed in Chapter 6 can be used in Human-Computer Interaction. As an example, an implementation to track visual attention and to identify and interpret pointing gestures in X3D in accordance with the W3C standard X3D is presented.

The chapter begins with a short introduction to the interaction concept realized in X3D. On this background, extensions implementing the models specified in Chapter 6 are discussed. While this is straightforward for low-level interactions, such as attention awareness and direct manipulations, the object-centered approach adopted by X3D seems to be less suited for high-level interactions. The presented implementation consequently adopts a user-centered approach.

A.1 X3D, InstantIO and InstantReality

The framework is implemented on top of instant**reality** (Fellner et al., 2009) developed by Fraunhofer IGD, Germany. instant**reality** is a browser for the eXtensible 3D (ISO 19775-1:2004, 2004) description language (X3D) for interactive 3D environments. X3D is primarily targeted at desktop environments and standard interaction devices such as mouse and keyboard. instant**reality** extends X3D to support augmented, mixed and virtual reality installations using state-of-the-art technology, both for input and output devices.

The presented framework stays within standard X3D whenever possible, so that most parts can be used in other X3D browsers as well. Only the basic access to the interaction devices, the eye tracker and the motion tracking

system, relies on the InstantIO system for networked device access provided by instant**reality**.

X3D Scenegraph Interactive 3D environments are described in X3D using a scenegraph. This is basically a hierarchical tree structure consisting of nodes (leaves) and group nodes (branches). In the scenegraph, effects are forwarded top-down. A common example is the TransformNode, a grouping node that can be used to position objects in the environment. This is achieved by changing the reference coordinate system for all of its children. The spatial layout of the virtual environment is typically realized using a hierarchy of such TransformNodes.

X3D Fields Nodes in X3D have properties, such as the translation of a TransformNode, that are accessible using X3D fields. Several types of field are supported, examples are fields for boolean values (SFBool), floating precision numbers (SFFloat), vectors (SFVec2f, SFVec3f), strings (SFString) or matrices (SFMatrix). Fields exist in two varieties, single fields holding one value, as described above, and multi fields holding arrays of values. The two kinds can be told apart by the first character, which is either S for single or M for multi fields (e.g. MFString). In the illustrations in this chapter, fields are depicted as rounded rectangles placed on top of their node (see Figure A.2 for an example). Fields also have an access type. They can either receive values (inputOnly), produce values (outputOnly) or do both (inputOutput). A special kind of field (initializeOnly) only receives an initial value, which cannot be routed (see next paragraph) and cannot be changed once the node is instantiated. Depending on the access type, the graphical representations of the fields in the illustrations are extending beyond the frame representing their parent node. If they extend to the left, they accept input and if they extend to the right, they produce output.

X3D Routes The scenegraph describes the structure of the virtual environment, which is rather static in most cases. The dynamic aspects of a virtual environment are expressed in a data-flow graph orthogonal to the scenegraph, by connecting individual fields using X3D routes. An X3D route links a *fromField* of a *fromNode* with a *toField* of a *toNode*. If the value in the *fromField* changes, the linked *toField* (or *toFields*, as there can be several routes defined) is changed accordingly, during a field propagation process that is executed in every application cycle.

The **TimeSensor** node, for example, produces updates of the time that has elapsed in terms of a fraction of the specified time cycle. This fraction has the type SFFloat that can be routed, for example, in the transparency field of a material definition to visually fade-out an object. Several other types of nodes exist that produce field updates, which are either driven intrinsically, as is the **TimeSensor**, or extrinsically, for example based on user interaction. In the illustrations, routes are depicted as dashed lines between fields (see Figure A.6), with an arrow indicating the flow of the data.

X3D Scripting More complex interactive behavior can be specified using X3D ScriptNodes, either in JavaScript or in Java. The interface of ScriptNodes can be defined by declaring a set of fields that operate as data channels between the scenegraph and the script. ScriptNodes can then be used as first-class scenegraph nodes, just as the sensor nodes defined in the X3D specification. Most of the functionality provided by the DRIVE framework is implemented using ScriptNodes.

InstantIO: Linking Devices to X3D InstantIO is a lightweight framework for networked device access. The basic component of InstantIO is a node. A node can be a driver for a specific device or a provider for auxiliary services. The networking functionality of InstantIO as well as a web-based management interface are implemented as auxiliary nodes. Nodes can provide typed input and output slots, analogous to the fields of X3D. Input and output slots can be linked using routes, similar to the procedure in X3D. Related slots can be organized in hierarchical namespaces, orthogonal to the parent-child relationship of nodes and slots. This flexible namespace system provides an automatic routing mechanism by connecting output slots with input slots of a corresponding fully qualified name (namespace plus field name, wildcards allowed). More details can be found in the InstantIO documentation (Dähne, 2009).

A.2 Device Access

A.2.1 Example of a Hardware Set-Up

The interconnection of the devices and the InstantIO nodes that were set up at our laboratory for this thesis are depicted in Figure A.1. The set-up consists of three computer systems.

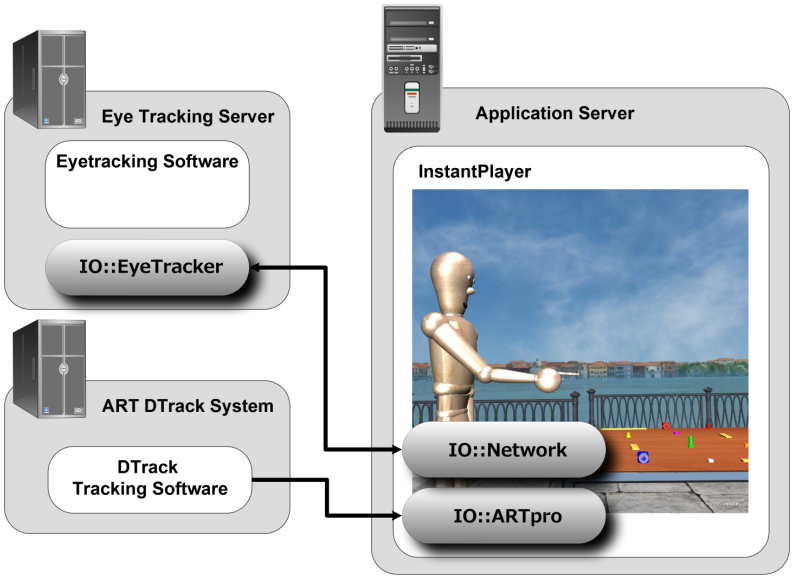


Figure A.1: *The device set-up for the integration of the tracking systems used in this chapter. An InstantIO node IO::EyeTracker controls the eye tracking software and publishes eye movements in the InstantIO namespace on the network. This data is collected by the IO::Network node in the process of the X3D browser InstantPlayer, and exposed via X3D fields for further processing. A data flow directed in the opposite direction allows the X3D browser to control the eye tracker, for example when calibrating. The motion tracking system is attached using a different scheme. The DTrack software broadcasts tracking data, which is collected by the IO::ARTpro node in the InstantPlayer.*

- The **Eye Tracker Server** is connected to the eye tracker. It runs the vendor-specific eye tracking software. An IO::EyeTracker node controls the eye tracking software using a vendor-specific API, and receives the current eye positions, which are published to an InstantIO namespace.

- Motion tracking is handled by the **ART DTrack System**. It is connected to the tracking cameras (not shown in the figure) and analyses their data using the DTrack software. The tracking data is broadcast via multicast or sent by the User Datagram Protocol (UDP).

- The **Application Server** hosts the application in the InstantPlayer X3D browser. It connects to the eye tracker using networked InstantIO,

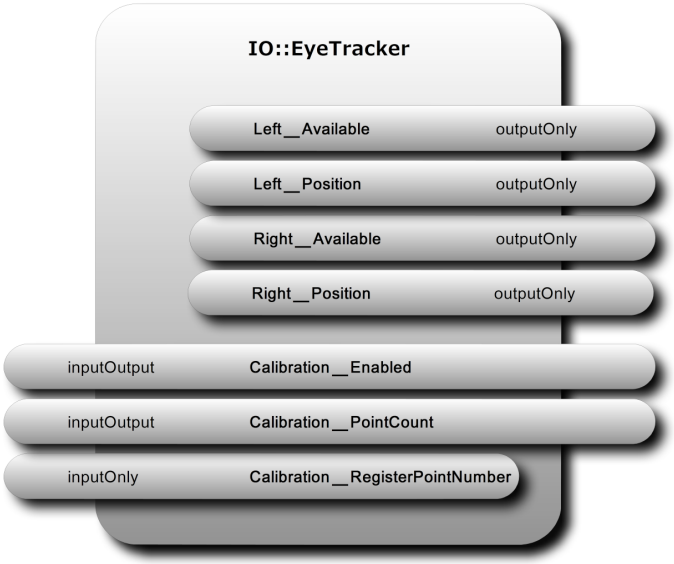


Figure A.2: *The InstantIO node IO::EyeTracker provides a generic interface to eye tracking systems. The first part of the interface provides information about the availability and the position of the eyes as output. A second part of the interface can be used to control the calibration procedure to parameterize the mapping between eye space coordinates and reference points on a 2D plane.*

and to the motion tracking using the IO::ARTpro node, which receives the data from the DTrack system via multicast or UDP.

A.2.2 Eye Tracking

The IO::EyeTracker node is shown in Figure A.2. The node uses a vendor-specific API to connect to the eye tracking software. Currently, models from Arrington Research and from SR Research (and early SMI devices) are supported. The IO::EyeTracker node publishes two output slots for each eye, **Left-/Right_Available** and **Left/Right_Position**. **Left/Right_Available** are boolean slots, which are set when the device is connected and data is received for the specific eye. In monocular settings, only one of the two slots can be true. The **Left/Right_Position** slots provide the current position of the eyes in normalized eye space coordinates. The coordinates in eye space are mapped to a 2D plane in a second step. For this mapping, an interactive calibration step is required.

The interactive calibration is controlled using the slots **Calibration_Enabled**, **Calibration_PointCount** and **Calibration_RegisterPointNumber**. **Calibration_Enabled** switches between calibration mode and normal operation. The number of reference points used in the calibration process can be specified using **Calibration_PointCount**. Typical counts are 9 or 16. A higher number of calibration points can improve accuracy, but the calibration procedure will take correspondingly longer. **Calibration_RegisterPointNumber** is set to the number of the reference point the user is actively focusing on. In the moment the number is set, the current eye position is taken as the feature vector for the current reference point, and the mapping is adjusted accordingly. If possible, the IO::EyeTracker node internally uses the calibration procedures provided by the vendors of the devices.

A.2.3 Motion Tracking

The IO::ARTpro node to access the ART tracking system is shown in Figure A.3. The ART system tracks targets with 6 degrees of freedom (6 DOF), that is, position and orientation. For each target, the IO::ARTpro node publishes the slots **Matrix *X***, **Position *X*** and **Orientation *X***, with ***X*** being the number of the target. The information is published redundantly for convenience.

The ART system also provides active targets for finger tracking. The main body of these targets is attached to the back of the hand, and the node provides position and orientation in the slots **H*X*_Matrix**, **H*X*_Position** and **H*X*_Orientation**. There are several different sets of finger targets, tracking either 3 or 5 fingers. The number of fingers currently being tracked is published in **H*X*_FingerCount**, and the position of the finger tips in **H*X*_*Finger*_Position**, where ***Finger*** can be thumb, index, middle, ring or pinky.

A.3 Detecting Gaze Pointing

The first step towards detection of gaze pointing is the detection of a fixation. There are several coordinate systems in which fixations can be detected. The first is the 2D eye space coordinate system, which is approximately a polar coordinate system residing in the center of the eyeball. There is also the 2D coordinate system used by the eye tracking system, which can either be similar or equal to the eye space coordinate system or it might be intrinsic

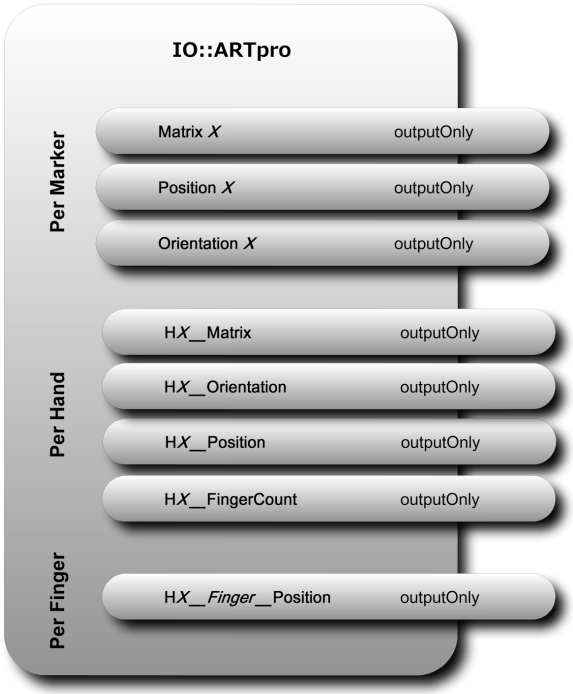


Figure A.3: *The InstantIO node IO::ARTpro provides access to the ART tracking system. Positions and orientations for a number of 6 DOF targets are held in the Matrix/Position/Orientation slots (X is the number of the target). Specific slots exist for each of the hand and finger tracking devices. For individual fingers, the position of the finger tip is provided in* HX_Finger_*Position.*

to the recording video camera. Then there is the 2D projected coordinate system of the 2D plane which the eye tracking system has been calibrated to. This is the coordinate system most eye tracking applications operate in, and most of the time its dimensions are given in terms of the pixel coordinates of a computer screen. Finally, there is at least one 3D coordinate system, the world coordinate system (real or virtual).

A.3.1 Detecting 2D Fixations

Several algorithms have been proposed to detect fixations in raw eye movement. Salvucci & Goldberg (2000) compared the most prominent representatives and came to the conclusion that HMM-based (hidden markov model) and

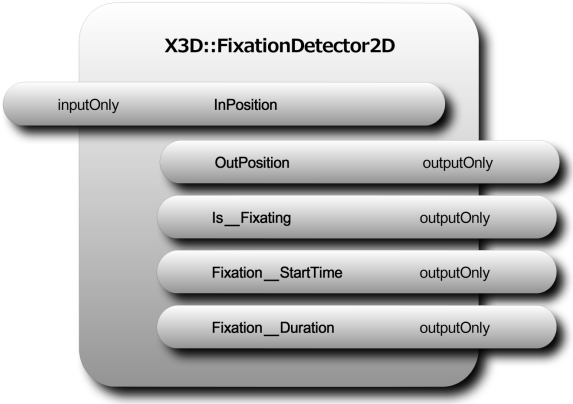


Figure A.4: *The X3D::FixationDetector2D detects fixations in eye space. In the typical case, its* ***InPosition*** *is linked to one eye's position field from the IO::EyeTracker node. If a fixation is detected, the boolean flag* ***Is_Fixating*** *is set. More details about the fixation are provided in the remaining fields.*

dispersion-based algorithms have a high accuracy. They are also very fast, which is the most important requirement for fixation detection algorithms used in human-computer interaction. An example implementation of I-DT written in Java according to the description in Salvucci & Goldberg (2000) is shown in Listing A.1. The algorithm is efficient and accurate for offline analysis, but is less suited for interactive systems. To give an example, the duration covered by the history of raw gaze points provided to the algorithm has to be longer than the maximum duration of the expected fixations. Thus a delay is induced before the algorithm will signal even the start of a fixation. In real-time interaction, the algorithm will also be evaluated at every application cycle, but in the original algorithm it is left unclear how such iterative applications should be handled.

The component X3D::FixationDetector2D (see Figure A.4) therefore implements a dispersion-based algorithm that has been especially designed for real-time interaction (see Listing A.2). In this implementation, called RI-DT, the current hypothesis of the existence of a fixation is updated per incoming raw gaze point, and no history is maintained. If an incoming raw gaze point stays within range of the dispersion threshold, the fixation hypothesis is updated accordingly. If the duration of the collected raw gaze points exceeds the minimum threshold, a fixation is detected and the corresponding flag _is_fixation is set. Once a raw gaze point lies outside the range, the state is reset and the search for a new fixation starts. In borderline cases, the

```
1
2  public Vector detect_fixations ( Vector raw_points, Vector timestamps )
3  {
4    Vector fixations = new Vector();
5    while( !raw_points.isEmpty() )
6    {
7      int index = 0;
8      float x_min = Float.MAX_VALUE, x_max = Float.MIN_VALUE;
9      float y_min = Float.MAX_VALUE, y_max = Float.MIN_VALUE;
10     Point center = new Point();
11     foreach( Point p : raw_points )
12     {
13       x_min = Math.min( x_min, p.x );
14       x_max = Math.max( x_max, p.x );
15       y_min = Math.min( y_min, p.y );
16       y_max = Math.max( y_max, p.y );
17
18       if( x_max - x_min > X_THRESHOLD || y_max - y_min > Y_THRESHOLD ) break;
19
20       center.x = (center.x * index + p.x) / (index + 1);
21       center.y = (center.y * index + p.y) / (index + 1);
22       index++;
23     }
24
25     if( timestamps.get(index) - timestamps.get(0) > MIN_FIXATION_DURATION )
26       fixations.add(
27         new Fixation()
28             .setCenter( center )
29             .setStart( timestamps.get(0) )
30             .setDuration( timestamps.get(index - 1) - timestamps.get(0) ));
31
32     raw_points.removeRange( 0, index );
33   } return fixations;
34 }
```

Listing A.1: *The I-DT dispersion-based algorithm for fixation detection. The function detect_fixations is called with the list of raw gaze points and associated timestamps measured by the eye tracker. The function returns a list of detected fixations.*

behavior of RI-DT differs slightly from I-DT. In RI-DT, if two or more raw gaze points have already been found within the dispersion thresholds, but the duration threshold has not yet been exceeded, a new raw gaze point that makes the total range exceed the dispersion threshold will start a completely new search for a fixation. In contrast, I-DT will first try to remove the first raw gaze point from the queue and test whether the remaining raw gaze points together with the new one stay in range. The frequency of these borderline cases depends on the X/Y thresholds, on the minimum fixation duration, which can be lowered to reduce the effects of this difference, as well as on the dispersion threshold, which can be raised to reduce the effects of this difference.

```
1
2  public void detect_fix_RT ( Point raw_point, double time ) {
3    _x_min = Math.min(_x_min, raw_point.x );
4    _x_max = Math.max( _x_max, raw_point.x );
5    _y_min = Math.min( _y_min, raw_point.y );
6    _y_max = Math.max( _y_max, raw_point.y );
7
8    if( _x_max - _x_min > X_THRESHOLD
9     || _y_max - _y_min > Y_THRESHOLD )
10   { // this point does not belong to the last fixation, start new one
11     _x_min = raw_point.x; _x_max = raw_point.x;
12     _y_min = raw_point.y; _y_max = raw_point.y;
13     _center.x = raw_point.x; _center.y = raw_point.y;
14     _index = 1; _start_time = time; _duration = 0;
15     _is_fixation = false;
16   } else {
17     _center.x = (_center.x * _index + raw_point.x) / (_index + 1);
18     _center.y = (_center.y * _index + raw_point.y) / (_index + 1);
19     _duration = time - _start_time;
20     _is_fixation = _duration > MIN_FIXATION_DURATION;
21     ++_index;
22   }
23 }
```

Listing A.2: *The RI-DT algorithm, a version of the I-DT algorithm optimized for real-time performance.*

For human-computer interaction, it is essential to distinguish between the different kinds of processes that drive eye gaze. Following fixations that are part of the visual search trajectory of a user may enable a system to guide the user in his search, for example by adding more specific constraints. If the user is producing an utterance, the fixations at the end of the visual search trajectory are most relevant, as they dwell with high probability on the object the user is going to talk about.

A relationship between the type of processing and the typical duration of the fixations it produces has been presented in Chapter 2. Figure A.5 presents a classifier that was built based on the findings described in Velichkovsky et al. (1997). It classifies fixations into the categories *figurative*, *semantic* and *communicative*, solely based on their durations. This classification allows the application to react to eye gaze on different levels of processing. The typical fixation durations for the different classes overlap, and consequently the classes are not disjunct. This may lead to transitions of fixations through different classes over time.

The processing components presented so far are linked to a short cascade for processing 2D eye gaze patterns, as depicted in Figure A.6.

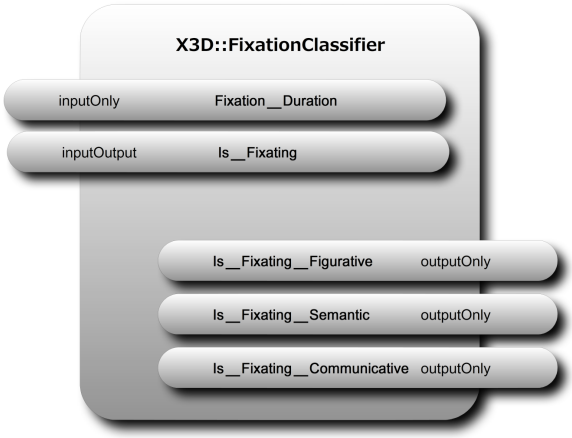


Figure A.5: *The component for classifying fixations according to the schema provided by Velichkovsky et al. (1997).*

A.3.2 Mapping Eye Movements to a 2D Projection Plane

Eye tracking systems are often used in computer-based studies, where stimuli are presented on a flat computer display. As a consequence, eye tracking systems often provide their gaze coordinates not in terms of the eye space, but mapped on a 2D plane, or even in pixel coordinates. This mapping needs to be calibrated before each use. The typical calibration procedure uses a sequence of fixation targets, commonly arranged in a grid of 9 or 16, that has to be followed by the user. This way, the systems can establish reference points for the mapping (see Figure A.7).

The IO::EyeTracker supports an application-controlled calibration procedure (see Figure A.2). The calibration module controls the presentation of fixation targets, here red spheres. During the calibration procedure, each of the spheres is highlighted once, and the calibration module waits for a detected fixation above a certain duration threshold. If the threshold is exceeded, the calibration module sets the **Calibration_RegisterPointNumber** slot of the IO::EyeTracker to the number of the currently fixated target. This is when the eye tracker associates the current orientation of the eye in eye space with the selected reference point. Alternatively to the automatic fixation detection during calibration, the procedure can also be completed using a self-paced iteration, for example controlled by a button. This way the calibration can be performed faster by a trained user.

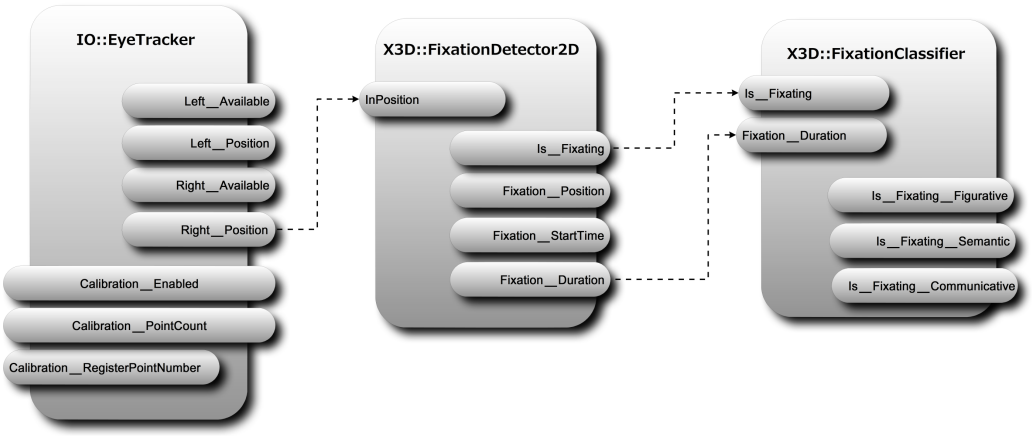


Figure A.6: *The short cascade of components for detecting 2D fixations. Information about the 2D position of the pupil in the eye coordinate system is used to identify 2D fixations. Based on the timing of the fixations, they are classified as being the product of different levels of visual processing.*

A.3.3 From 2D Positions to 3D Directions

To move from 2D gaze positions (either in eye space or projection space) to 3D gaze directions along the visual axes of the eyes, the position and orientation of the head of the user has to be taken into account, too. Technically, this means that information from two sensor devices needs to be fused. Figure A.8 shows the wiring diagram of the essential components realizing the transition from 2D to 3D.

The component X3D::GazeDirection provides the origin of the gaze direction, which coincides with the center of the eye, and the direction along the visual axes. To calculate both, it requires the current position and orientation of the point between the eyes (**Between_Eyes_Matrix**), which is provided by the motion tracking system, and the gaze position on the 2D projection plane (**Gaze_Position_2D**). In addition, the distances between the center of the eye and the position between the eyes (**Eye_Separation**), and the transformation between the center of the eyes and the projection plane (**Eye_Plane_Transform**) have to be specified.

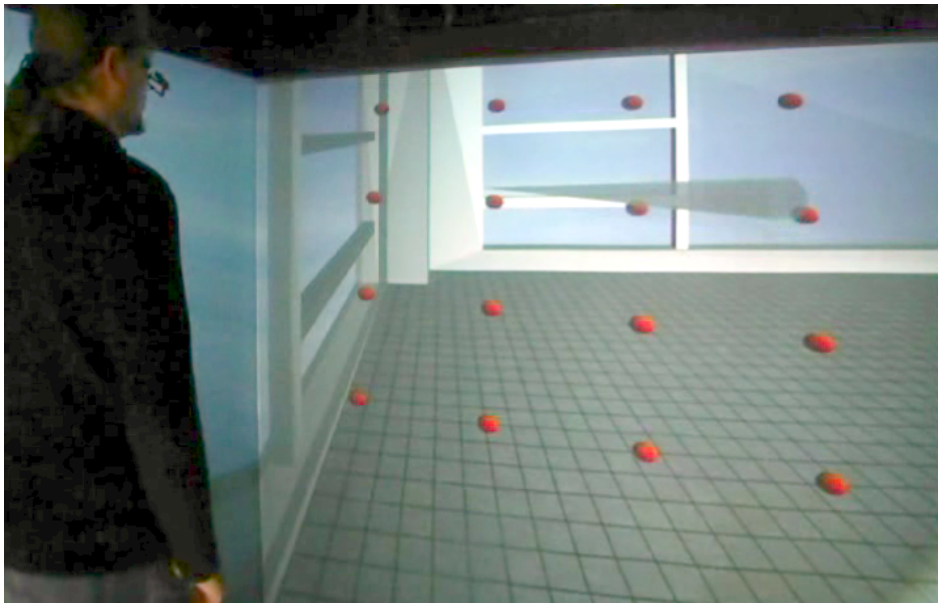


Figure A.7: *For the calibration process, the user has to iterate over a grid of fixation targets. The eye tracking system uses these fixations as references for the mapping between eye space and projection space.*

A.3.4 Detecting 3D Fixations

For the detection of the position of the point of regard in 3D, a triangulation using the visual axes of both eyes can be made, as described in Chapter 3. Figure A.9 shows the corresponding component X3D::Triangulation. The advantage of this approach is that it is fast and does not require additional calibration. The accuracy of the triangulation crucially depends on the accuracy of both visual axes, and small divergences lead to a degraded accuracy. Data on the accuracy and precision achieved in a study on estimating the 3D point of regard has been presented in Chapter 6.

A second component, X3D::MultiTriangulation (see Figure A.9), extends this approach to multiple axes. The idea is that small shifts of the user's perspective, which happen all the time in ongoing interaction, can be used to estimate the 3D position of the point of regard. The accuracy of this approach improves, the longer a user fixates a specific position in 3D space, and the more the user moves perpendicular to the visual axes during that time.

The PSOM approach to estimate the 3D position of the point of regard is implemented in the component X3D::PointOfRegard3DPSOM. Chapter 5.6 showed that the PSOM approach provides better accuracy and precision

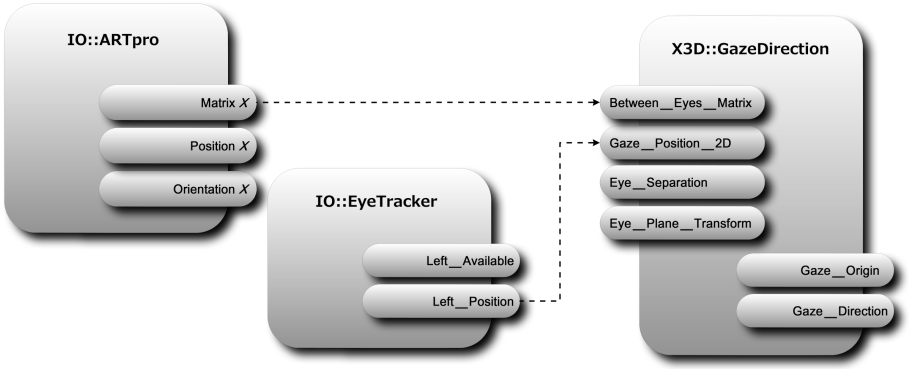


Figure A.8: *Information from the eye tracker (eye) and the tracking system (head) has to be integrated to construct the position and gaze direction of the eye in 3D space. For the two IO components, only a subset of the field interfaces are depicted for clarity.*

than the approaches based on triangulation. A drawback is the duration of the initial calibration procedure for the PSOM, which is longer than 2D calibration. The calibration procedure is triggered using the interface shown in Figure A.10. The calibration procedure is similar to that used for 2D calibration, but instead of a 2D grid of 9 to 16 points a 3D cubic grid of at least 27 points is used.

A.4 Detecting Manual Pointing

While manual pointing gestures show a great variety, the components described in the following only detect ideal manual pointing gestures. Ideal manual pointing gestures follow the description given in Section 2.3, that is, the handshape shows an extended index finger, the remaining fingers are curled towards the palm of the hand, and the movement of the hand has a clear climax during the stroke. The small and robust detectors developed for this thesis can be easily exchanged with high-quality detectors, which could, for example, detect the curling of the fingers based on the angles of the finger joints, instead of the detection based on the positions of the finger tips.

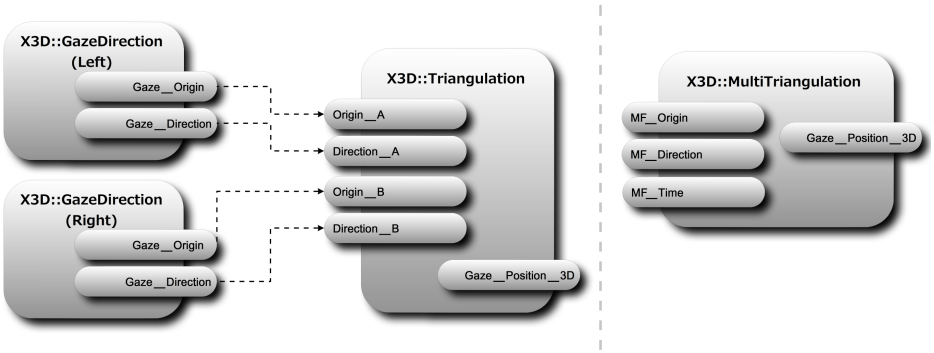


Figure A.9: *The X3D::Triangulation component to the left implements the triangulation method described in Chapter 3 to estimate the 3D position of the point of regard in space by intersecting the two visual axes calculated in the X3D::GazeDirection components for the left and the right eye. The X3D::MultiTriangulation component extends this approach to a triangulation of multiple (2 or more) axes.*

```
(define is-pointing-shape?
  (and  (>  (distance-current index-finger-tip hand-back)
            (* 0.85 (distance-max index-finger-tip hand-back)))
        (<  (distance-current middle-finger-tip hand-back)
            (* 0.75 (distance-max middle-finger-tip hand-back)))
        (<  (distance-current ring-finger-tip hand-back)
            (* 0.75 (distance-max ring-finger-tip hand-back)))
        (<  (distance-current pinky-finger-tip hand-back)
            (* 0.75 (distance-max pinky-finger-tip hand-back)))))
```

Listing A.3: *The handshape of a pointing hand is detected using a set of simple features.*

A.4.1 Detecting the Handshape

The X3D::HandshapeDetector (see Figure A.11) uses the position and orientation of the hand as well as the available positions of the finger tips (3 or 5 fingers) to detect a pointing with the index finger. As shown in Listing A.3, the detector identifies a pointing shape if the index finger is extended to more than 85% of its maximum extension and the remaining fingers curl to no more than 75% of their maximum extension. The position of the thumb is ignored, as the information of the curling of the remaining fingers is sufficient, and in praxis the optical detection of the position of the thumb is difficult if the thumb is hidden inside the palm.

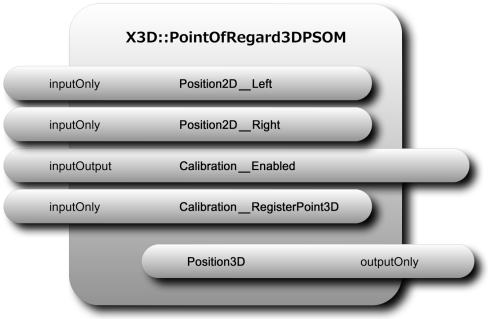


Figure A.10: *The X3D::PointOfRegard3DPSOM component implements the PSOM algorithm to estimate the 3D position of the point of regard using machine learning.*

A.4.2 Detecting the Stroke

The detection of the stroke is based on the velocity profile of the back of the hand and the handshape detection (see Figure A.12). The basis of the calculations is the direction of the pointing handshape. It defines the axis along which in a second step the velocity of the back of the hand is calculated. A stroke is detected if the velocity decreases from medium/high to about zero, and the stroke is aborted if the velocity decreases (retraction) again.

A.5 Interpreting Pointing

With the components presented so far in this chapter, manual and gaze pointing gestures can be detected. The interesting part now is how to determine the extension of the pointing gestures. This will be shown in the following.

A.5.1 Defining the Pointing Domain

First of all, the entities that could in general be the targets of a pointing gesture, the pointing domain, have to be made known to the system. The nature of the entities depends on the application domain. In the examples used for the experiments (see Chapters 4 and 5), the entities in the pointing domain were toy building blocks, each of which was represented in the scenegraph as a group of X3D nodes for material, position or shape. Other applications

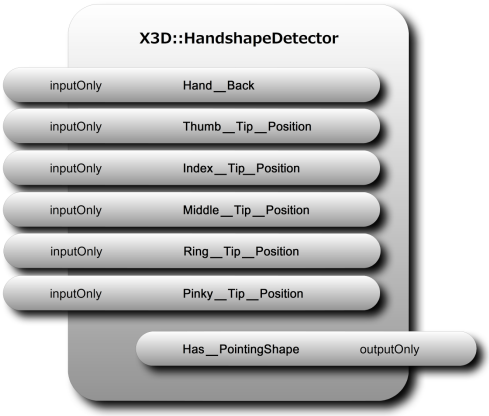


Figure A.11: *The X3D::HandshapeDetector component implements a basic handshape detection on the basis of the positions of 3 to 5 finger tips and the back of the hand.*

could require a more fine-grained pointing domain, for example in interactive geometry modeling, when individual vertices on a sub-node level are the subject of the user's investigation. The approach developed in this chapter is restricted to the case described first, to entities that are described as nodes in the X3D scenegraph.

The X3D standard provides Metadata nodes to support application-specific extensions. Each X3D node can be the parent of a set of such Metadata nodes. We will use Metadata nodes to add semantic annotations to existing nodes in the scenegraph, marking them as entities that can be the target of deictic expressions. This idea adopts the use of semantic entities (Biermann et al., 2002) used to annotate virtual reality scenegraphs in the research tradition of intelligent computer graphics. In this special case, the relevant entities will be annotated with a MetadataSet node with the name **Object** and the special property **is-perceivable** (see Listing A.4). As X3D does not provide boolean metadata, an integer type is used and the usual convention where a value of 0 stands for false and all other values stand for true is followed.

Technically speaking, deictic expressions are similar to so-called picking operations. Picking is the process of detecting intersections of a geometric model of a user interface tool with object geometries. In case of success, a set of objects that have been "picked" is returned. The X3D standard provides a specialized set of nodes specifically geared to picking operations. First, pickable objects can be marked with a special group node **PickableGroup**. Second, there are several sensor nodes (**LinePickSensor**, **PointPickSensor**, **PrimitivePick-**

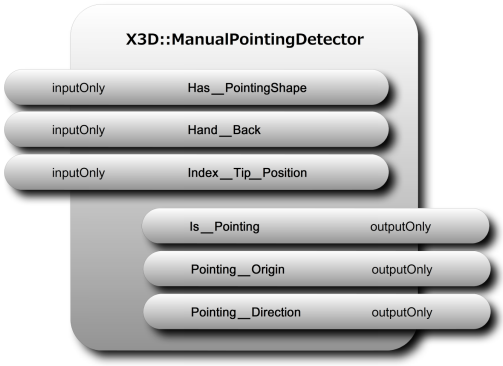


Figure A.12: *The X3D::ManualPointingDetector component identifies the stroke of a pointing gesture and calculates the origin and direction of a pointing ray when a manual pointing stroke is detected. The pointing ray is the basis for many dereferencing mechanisms.*

```
<Group DEF='Box-52'>
  <Shape><Box size='1 1 1'/></Shape>
  <MetadataSet name='Object'>
    <MetadataString name='is-a' value='box' containerField='value'/>
    <MetadataInteger name='is-perceivable' value='1' containerField='value'/>
  </MetadataSet>
</Group>
```

Listing A.4: *An annotation of a geometry subgraph describing it as an* ***Object*** *that can be the target of deictic expressions.*

Sensor and **VolumePickSensor**) that are specialized for different geometry models for picking. Using these nodes, the basic functionality of identifying relevant entities can be achieved using X3D standard nodes. The advantage of the X3D picking system thereby is its fast implementation, faster than any implementation based on the metadata nodes can be, simply because the picking system can prune irrelevant geometries early in the process. The X3D picking system is therefore used internally, whenever possible, as one building block of the deixis dereferencing system.

The X3D nodes make no assumption about the kind or parameterization of the model used for dereferencing. This parameterization is provided by the developed framework. In addition, the framework goes beyond the basic functionality in some aspects:

- Identifying pickable objects requires them to be grouped below a **PickableGroup**. This either has to be done when designing the content, or the scenegraph needs to be rearranged during runtime, as the **PickableGroup** has to be inserted within the existing hierarchy. The method based on the metadata nodes only requires nodes to be added as siblings, leaving the original structure intact.
- Picking in X3D only works within one context, it does not, for example, cross inlining of subgraphs. The use of picking thus has an impact on the overall design of the scenegraph.

A.5.2 General Interface Description

The deixis dereferencing components all exhibit a common output interface. The set of possible referent entities – the extension – is provided as MFNode field **Extension**, along with a rating between $-\infty$ and 0 in the MFFloat field **Ratings** for each entity. The rating expresses the quality of the match. A rating of 0 describes a perfect match and less likely matches receive ratings below 0.

A.5.3 Vector-based Dereferencer

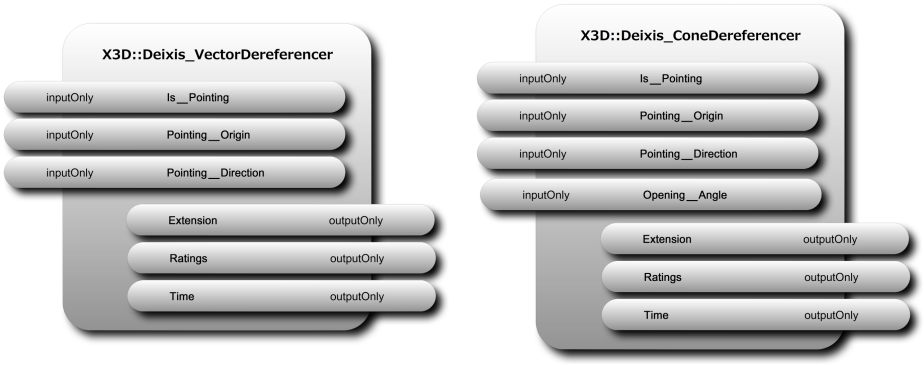


Figure A.13: *The dereferencers construct the extension of a deictic reference based on a parameterized reference model. This diagram shows two basic dereferencers, the Deixis_VectorDereferencer, which casts a ray from the specified origin along the direction, and the Deixis_ConeDereferencer, which uses a cone with a specified opening angle instead. The dereferencers provide the set of potential referents, the extension, and a rating between* $-\infty$ *and* 0 *assessing the likelihood of the potential referent.*

The X3D::Deixis_VectorDereferencer component (see Figure A.13, left) implements the naive ray intersection test to determine possible referent entities. The **Pointing_Origin** and **Pointing_Direction** of the ray can be externally defined, for example by routing from the output of the X3D::ManualPointing-Detector for manual pointing gestures. The evaluation in Chapter 6 showed, that the GFP/dom model provides the best approximation of the direction of manual pointing. Entities that are hit by the ray are listed in the field **Extension**. Several algorithms have been developed to calculate the ratings:

- **distance**: the negative distance from the origin of the pointing to the intersection point is used as a rating score. Touching entities at the pointing's origin gives the highest rating (near 0).
- **ranked-distance**: this is similar to **distance**, but the distance is expressed between -1 and 0. The first entity being hit has the rating 0, the last entity has rating -1.
- **normalized-distance**: this is similar to **distance**, but the distance is expressed between -1 and 0, with 0 being at the origin of the pointing and -1 being at the end of the pointing vector. For this rating algorithm, the length of the vector needs to be set in an additional field.
- **bbox-centrality**: the entities are rated according to the angular distance between the pointing vector and a vector from the pointing origin towards the center of the bounding box of the object. The smaller the angle between the two vectors, the higher the ranking. This algorithm prefers hitting objects in their center.

This deixis dereferencer based on vector extrapolation is, as has been shown in the experiments on manual pointing (see Chapter 4 and especially Chapter 6), not suitable for conversational interfaces without direct system feedback. The precision and accuracy of unguided pointing is too low. For direct human-computer interaction, where visual feedback such as a visual beam shooting from the pointing hand's index finger is provided as an aiming aid, the algorithms provide heuristics to differentiate between multiple entities that are intersected by the ray.

A.5.4 Cone-based Dereferencer

For manual pointing, a cone-based dereferencing model can be used to approximate the human's pointing. The corresponding component is depicted in

Figure A.13, on the right. It implements the pointing cone model described in Section 6.3.2.

The parameters **Pointing_Origin** and **Pointing_Direction** should be chosen according to the description of the vector-based dereferencer. In addition, the cone-based dereferencer also requires the opening angle of the cone to be set in the field **Opening_Angle**. Suitable parameters for the opening angle can be found in the evaluation of different opening angles presented in Section 6.3.2. The **Ratings** are calculated according to the weighting function based on orthogonal errors presented in the same section.

A.5.5 Hybrid Dereferencer

The **HybridDereferencer** is depicted in Figure A.14. It implements the hybrid pointing model described in Section 6.3.3. The hybrid dereferencer is basically a cone-based dereferencer as described above. In addition, an evaluation of the orthogonal error for the proximal area is added. For this evaluation, the interface of the cone-based dereferencer has been extended by two additional fields in the hybrid dereferencer. **Max_Orthogonal_Distance** delimits the maximum distance from the pointing ray an object might have. This restriction is not discussed in the section on the hybrid model. Nevertheless, it provides practical performance improvements in applications, as it allows the application to restrict the evaluation of possible referents to a smaller volume around the pointing ray. Typical values are 20 cm to 1 m, depending on the setting. The second field, **Proximal_Distal_Border**, can be used to specify the border between proximal and distal pointing. For distal pointing, only the pointing cone model will be used, as described in Section 6.3.3. In the proximal area, the orthogonal error will be measured in addition to the pointing cone. In the study on manual pointing presented in Chapter 4, the border between proximal and distal pointing was at about 40 cm.

A.5.6 Point of Regard Dereferencer

The component shown in Figure A.15 was developed in order to dereference gaze pointing with a measured 3D point of regard. It implements the model for location-based gaze pointing described in Section 6.4. To this end, the position of the **PointOfRegard** and the **EyePosition** can be specified, from which the model will be parameterized.

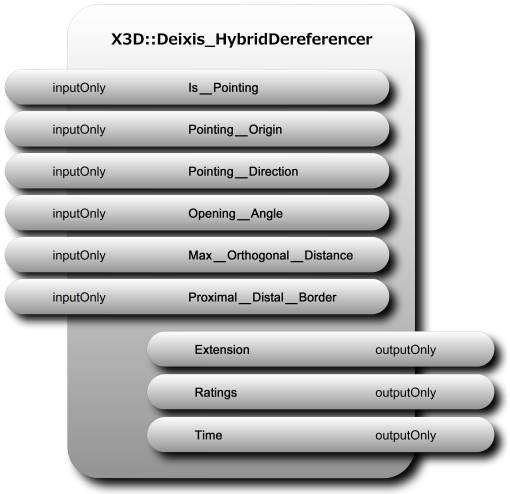


Figure A.14: *The Deixis_HybridDereferencer combines a Deixis_ConeDereferencer with a weighting function for the orthogonal error within the proximal area. A detailed description of the model implemented in this dereferencer is provided in Section 6.3.3.*

A.5.7 Interaction History

So far, the aspect of the asynchronous timing of multimodal interaction, with gaze preceding pointing, has been deliberately ignored in the presentation of the components, to reduce the complexity of the interface and to concentrate on the essential operations. Section 6.5 presents an example, where information about timing is necessary to arbitrate between manual pointing to an object-1 and manual pointing to an object-2. In this chapter, only the relevant aspects of the component interfaces for just-in-time processing are described. At any particular moment, the components and their fields present the current state of the interpretation of the ongoing user interaction. Yet some events of user interactions can only be correctly identified once they have been observed for a period of time or once they are already over. Also, as pointed out in Chapter 3, when bringing together different modalities, it will be necessary to detect patterns of asynchronous events, such as a gaze fixation preceding the stroke of a manual pointing gesture.

All components therefore also provide an additional multi-field for each of their fields, storing a history of previous field values. Each component also has one field of type MFTime called HistoryIndex, which stores the timestamps at which the other values in the multi-fields were valid and which serves as an index to those fields. If a value at a specific point in time has to be

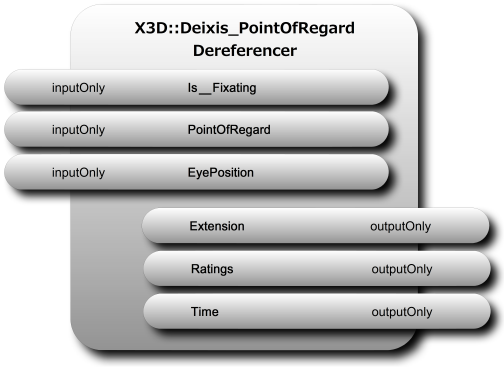


Figure A.15: *The Deixis_PointOfRegardDereferencer uses a gaussian distribution to model the extension of the point of regard in 3D as a function of its distance from the eye.*

retrieved, the HistoryIndex can be searched for the best matching timestamp, and the index of this timestamp can be used as an index to the desired history of values. Each component also has a field HistoryDuration, which can be used to restrict the maximum length of the histories to an appropriate value. Typically, components processing on lower levels require only short histories, while components on higher levels, such as the components providing object references, will have histories with durations of several seconds.

A.6 Summary

This chapter presented the DRIVE framework for deictic reference using gaze and gesture in virtual environments, which was developed in this thesis. The X3D standard and instant**reality** were chosen as a target platform, but the general concept is not restricted to this platform. A previous version of the interaction framework has been implemented for Avango/Performer as well. The framework embraces all aspects of the interaction processing, starting with accessing hardware devices, over detecting features, identifying gestures, to finally interpreting the extension of pointing gestures. The presentation in this chapter followed this data flow. Ultimately, the results are provided to the reference resolution system of a conversational interface, as described in Section 6.5. Through its component-based architecture, subgraphs of the interaction framework can be used for other aspects of natural communication as well. They have, for example, been used to inform turn-taking and to monitor joint attention processes (Pfeiffer-Lessmann & Wachsmuth, 2008).