

Interactive Graphics Design with Situated Agents

Ipke Wachsmuth and Yong Cao
Faculty of Technology, University of Bielefeld,
P.O. Box 100131, 33501 Bielefeld, Germany

1 Introduction

Along with sophisticated techniques for natural visualization and rapidly increasing power of modern graphics workstations, high-quality 3D graphics is becoming most attractive for design and simulation. One area in which this new media proves especially useful is architecture and interior design. For example, the visualization of an office room or a building prior to its physical realization could help a designer to obtain realistic impressions of a construction while it is evolving and give free way to imagination at the same time. It is one of the aims that, eventually, a designer is able to explore, and interact with, a manipulable environment without wasting physical matter and with the ability to readily change the immaterial model.

In this paper, a scenario for one of the projects in a new research program on “Artificial Intelligence and Computer Graphics” at the University of Bielefeld is described. In the VIENA project (“Virtual Environments and Agents”) we want to provide a way of intelligent communication with a technical system for designing and generating 3D computer graphics¹. To do so we apply new AI methods and techniques that build on ideas of situated communication and agents. We think of agents as mediating systems which cooperate with a user by exploiting internal scene information not readily available to the user. For instance, specialized agents can take on particular jobs with respect to geometry or material manipulation. To master communication with the user (designer), the mediating agents are informed about the actual scene as it is seen from the perspective of the user. By moving the camera it is possible to place the designer’s eye in different perspectives. This way, the designer is situated in the developing scene and can issue commands from his or her current perspective.

In the following section, we focus on current efforts attempting to bridge the gap between high-quality visualization and interactive systems. We argue that interaction modalities include language and symbols as means of communication. In Section 3, we point out how situated communication could help in interactive graphics design. An important issue is that the system has to know about the spatial structure as it is perceived and experienced by the human user. In Section 4, we sketch various

¹Research in the VIENA Project is partly supported by the Ministry of Science and Research of the federal state North-Rhine-Westphalia under grant no. IVA3-107 007 93. The authors are indebted to Norbert Siekmann, Britta Lenzmann, Majid Amanzadeh, Tanja Jörding, Stefan Fischer Rivera, and Karsten Otto for assisting during the research.

notions of agents that are currently found in literature, and we explain further the idea of intelligent mediation by way of situated agents. In Section 5, we give more details about our scenario and include a brief overview of a multi-agent system that we develop for the mediation of verbal instructions. The current state of our work is also described in this section. We conclude by arguing that artificial intelligence techniques can provide a communication link between humans and multimedia and also merge into more immersive environments.

2 Interactive Graphics Design

Not far ago, the output of conventional graphics systems was exclusively thought for viewing by a human user, that is, there was little possibility to interact with the displayed images. A major goal of current efforts is to bridge the gap between high-quality visualization systems which present output to a passive user, and interactive systems which are able to accept and display user interventions as soon as they are issued. In the ideal case, the user is immersed in a scene and is able to interact with objects in the scene. Examples have been given (e.g., Brooks 1987; Krüger 1993) how such a setting can be used for a three-dimensional pre-exploration of building construction-plans and interior design.

Accurate and natural visualization is important for obtaining a realistic impression of an evolving design. To give an idea, Figure 1 shows a synthetic scene of our lab interior which we use for exploring object arrangement, materials, and illumination. On the other hand, interactivity is an important asset to support a creative process in graphics design. Among most urgent research questions for virtual environments, an enhancement of software techniques for modeling has been called for (Bishop *et al.* 1992). In this context, modeling refers to the data structures which are used to record the scene information for a synthetic environment. These data structures describe the shape of the objects, their parts and physical properties, and how they interact with other objects in the environment and with the user.

While progress has still to be made before natural visualization can be used other than with largely precomputed models, the issue of modeling has its own difficulties and challenges. In manipulating a model, a designer needs to communicate with a technical device and may face crucial obstacles in the ease how the model is arranged and changed. A comfortable human-computer interface can keep the designer free from technical considerations such as planning of geometric detail. The user should be able to move through the model and change it interactively. At least part of physics should be in effect, for instance, it should not be possible for objects to pass through one another.

Some researchers have begun to use the data glove for rearranging objects in a scene (Böhm *et al.* 1992). To a human, it seems more natural to grasp a chair, lift it from the ground, and put it down at a new position, than to calculate an exact target position for changing the geometric model. The advantages of such a direct object handling – at least as it pertains to spatial manipulation – need not be mentioned. But changing the material (e.g., the color) or the size of an object would most likely involve a mental detour, for example, a “space menu” might have to be used.

We think that so-far available means of interaction, e.g., the data glove, are but one way to manipulate the arrangement of a scene. An alternative we explore in the VIENA project is to use verbally communicated changes which are put in effect by a mediating system. Interaction modalities would be further enhanced if designers could use *language* and *symbols* as a means of communication. Thus, we want to instruct the system to carry out changes where gestural manipulation is impossible or unnatural. Eventually, voice input and gestures could be used as parallel input modalities.



Fig. 1. A manipulable 3D object scene

3 Situated Communication

When using verbal interaction in geometry modeling, we need to be aware of the fact that the way we refer to details in a scene is “situated.” For instance, it may depend on the objects themselves where we would speak of “front” and “back,” e.g., a chair and a desk impose local reference structures on space, and they may have opposite “front” parts (Levelt 1986; *cf.* Figure 2). We may use still different notions of “front” and “back” – and also of “left” and “right” – when making reference to our current aspect of view. Thus, the system has to know about the spatial structure as it is perceived and experienced by the human user.

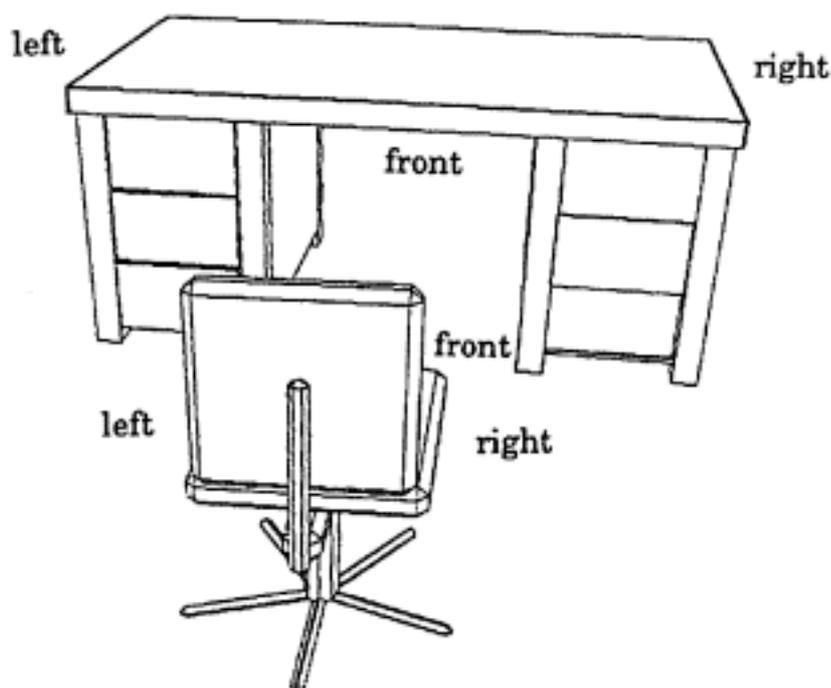


Fig. 2. Different use of particles ‘left’, ‘front’ and ‘right’ in a scene with a chair and a desk (after Levelt 1986)

To give another example, the three instructions, “move the table here, more to the front, more to the wall” incorporate different frames of spatial reference. In the first case, reference refers to the speaker’s position while it is anchored externally in the second and the third case. The metrical structure does not only depend on geometry but also, for example, on how far one could reach from a position. Thus the system – as the human – has first to find out which frame of reference is relevant, secondly, in which direction the table is to be moved, and thirdly, how far. It should not be possible to move an object further than permitted by a physical boundary. When the table is moved, the things on it should move with it (Fig. 3).

When a deictic reference is involved (“here”), the point of anchoring a reference frame depends on the current position of the speaker which could be identified with the camera position in a virtual environment (Fig. 4). In Fig. 4, 5, and 6 we show an observer of the scene to indicate the position of the camera. The current position of the observer (camera) can help to resolve ambiguity in instructions; e.g., “put the chair in front of the table” can be interpreted in different ways depending on the speaker’s position (Fig. 5). Finally, the environment is modified by the objects present in that an object can impose its own reference scheme. For example, the phrase “in front of” is most likely interpreted in different ways – by a human, and should be so by the system – in the scenes shown in Fig. 5 and 6.

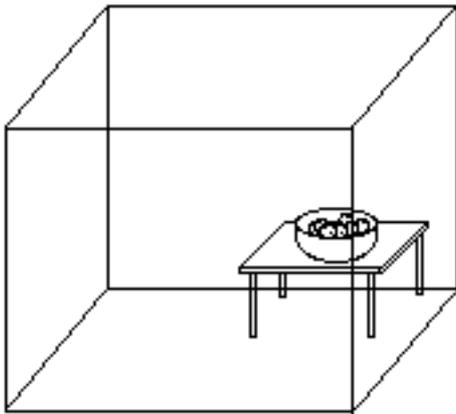


Fig. 3. “Move the table to the front!”
(What happens to the bowl?)

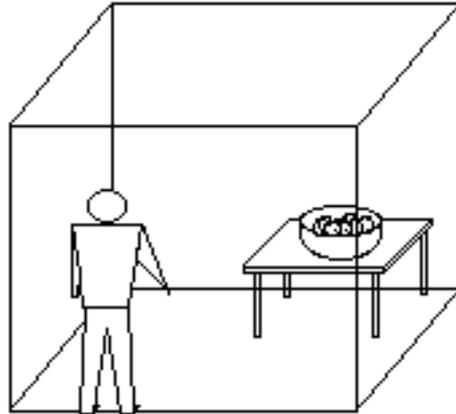


Fig. 4. “Move the table here!” (Deictic reference includes speaker’s position)

We want to exploit findings from cognitive research to deal with these topics. For instance, object schemata have been proposed (Lang 1989) that describe how space is modified by the objects present. The human visual-perceptive system gives rise to spatial relations between object and the observer’s body (Bryant 1991) which could be mapped onto an implicit camera. Different frames of reference were identified: (1) the egocentric frame, defined by the three body axes head/foot, front/back, and left/right; (2) the allocentric frame, defined by orthogonal axes independent of the observer. Such axes can be anchored in a prominent landmark in the environment or be oriented according to global directions (Cao 1993). In gravity, head/foot is identified with top/down when the observer is in upright position. Axes are experienced differently in other environments, e.g., in zero gravity (Friederici 1989).

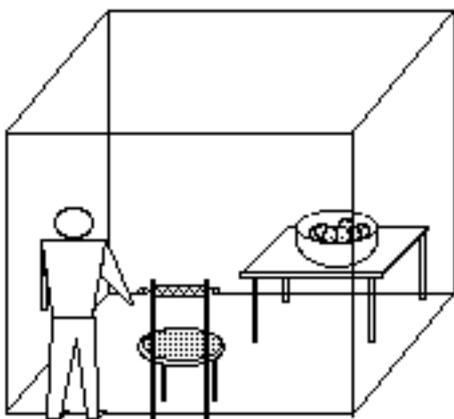


Fig. 5. “Put the chair in front of the table!”
(Modification of space by speaker’s position)

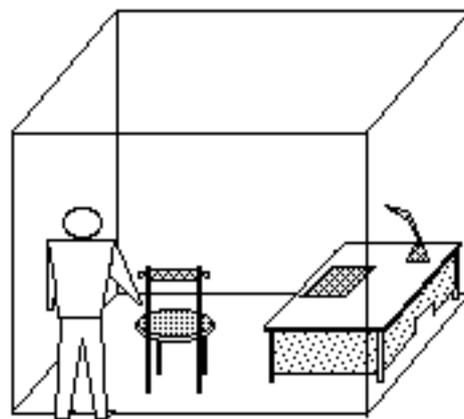


Fig. 6. “Put the chair in front of the desk!”
(Modification of space by objects present)

4 Agents

One theme in the mentioned research agenda (Bishop *et al.* 1992) is the development of knowledge-based agents for human-computer interfaces. An agent is an entity consisting of a structural definition, a set of functional units that defines its behavior repertoire, and some means for selecting and sequencing (possibly concurrent) behaviors (Zeltzer 1991). In our approach, we use a number of specialized agents to conceptualize and realize a system interfacing between verbal user instructions and a technical system for modeling/rendering. Before we explain how we go about this, we briefly sketch agent notions found in the current literature.

Ideas of a general intelligent agent that uses knowledge to perform actions in the service of goals have been researched in AI for more than twenty years (*cf.* Newell 1981). A novel perspective was taken by Minsky (1986) in explaining intelligence as a combination of many simple processes he refers to as agents. Recent attempts to develop more complex knowledge-based systems have revealed shortcomings of centralized, single-agent architectures and have acted as a springboard for research in Distributed AI (Adler *et al.* 1992).

Various models and notions of agents are currently found in diverse literature in the fields of autonomous and distributed systems (Meinkoehn and Knoll 1993), language-action systems (Chapman 1991), and graphics animation (Badler *et al.* 1991). Of interest, in our context, is work on “interface agents” (Laurel 1990; Maes and Kozierok 1993) which typically use knowledge about tasks, habits, preferences of their users to perform actions on their behalf. Such an idea has been adopted in visions of future user interfaces, for instance, in the Japanese Friend21 project and in the Newton project of Apple (*cf.* Marcus 1993).

On the other hand, more recent work in Distributed AI incorporates approaches for “cooperating agents” that work together in achieving an overall task. Multi-agent systems as discussed in (Steiner *et al.* 1990) emphasize the aspect of task-related cooperation of independent (autonomous) systems. Each agent is ascribed a basic functionality (it can solve certain problems), a cooperative head (for participating in a cooperation with other agents), and communication abilities (by way of accessing communication channels to other agents).

In particular, the idea of “situated agents” which can gain and exploit information from an actual situation is of interest to us. A situated agent integrates aspects of perception, action, and communication in one system to succeed in a situation without having a complete model of it (Brooks 1991). The term “situatedness” refers to the ability of an intelligent system to exploit the actual situation, to the extent possible, as a source of information in perceiving and manipulating its environment and communicating with cooperating partners.²

In our work, we conceive a situated agent as an “intelligent mediator.” Such an agent – which may consist of several subagents – communicates and cooperates with a human user in an overlapping perceptual situation. The key idea is to have an

²Situated communication is a focus theme in a newly established special research unit at the University of Bielefeld (SFB 360, “Situated Artificial Communicators”).

agent inspect the internal description of a scene that the user can experience by eye inspection, so both of them – agent and user – can communicate about scene details from their “point of view.” But whereas the user is likely to have a qualitative grasp of the scene, the agent has knowledge of exact object locations, colorings, etc. Qualitative communications of the user to change scene details will have to be evaluated by the agent(s) so as to produce appropriate quantitative changes in the scene description. The altered scene description is then visualized for the user. We refer to this overall process – which will be explained in more detail below – as “intelligent mediation.”

5 Agent-mediated user interaction in VIENA

In an example application of interior design, the overall goal of the VIENA Project is to enable and establish an intelligent communication with a technical device for the interactive design and exploration of 3D computer graphics. Our specific scenario is as follows: In a modeling session, the designer keeps track of the evolving design by viewing it in a 3D display setting. The designer can change the model by communicating with the system via simple verbal instructions (e.g., “move the table more to the front”). The system offers a view of a resulting scene where “more” is interpreted on the basis of a default value. The offer can be changed in further interaction (“still more”, “not that far”). In other words, computing the semantics of an instruction is a situated activity which leads to a scene modification, and the user can negotiate the semantics of an instruction.

To realize the mediation of verbal instructions, we have constructed a multi-agent system. As an ‘agency’ this system communicates with the human user (designer) to receive and process verbal instructions. Internally, agents cooperate with each other to realize the user’s instructions with respect to the current situation. We conceive agents as systems with restricted ability they bring to bear with respect to a given instruction. Some agents carry out spatial inferences to meet the expectations of the human user. In doing so, agents exploit the current situation to the maximum extent possible. By using information about the most previous manipulations, possible ambiguity in an instruction can often be resolved. Other agents know how to find out current locations and materials of objects, still others about how to change a color or an illumination (“darker!”), etc. The more competent, by agent mediation, the system becomes, the more successful the designer’s instructions can be interpreted and executed.

Overall System Description

In this subsection we give a brief overview of the VIENA system. Our current working environment includes a Silicon Graphics Indigo ELAN R4000 for the main project demonstrator. This machine supports the real-time hardware shading we make use of for fast scene visualization. Currently, we use SOFTIMAGE for scene modeling and rendering. Besides this, several Sun SPARC stations are available for

algorithm and model development. These are also the site for some of the agents, as we make use of distributed processing.

We have specified a client-server architecture where agents are autonomous processes which can send and receive messages by interprocess communication. For experimental implementation, we use two – exchangeable – realizations: (a) communication files in the Unix Network File System; (b) a low-level but fast realization with communication via sockets. We are also exploring higher-level protocols based on Remote Procedure Call for an improved realization of the specified communication system.

Our system acts as an interface between a human user and a 3D graphics modeler/renderer (*cf.* Figure 7). The user communicates changes to the system by way of verbal instructions. A Parser agent translates a user instruction to a structured representation which outputs to the mediating agents. Parser asks back if an incomplete instruction cannot be resolved by the agents themselves (or if it cannot be parsed). The user observes changes from the Viewing graphical output medium.

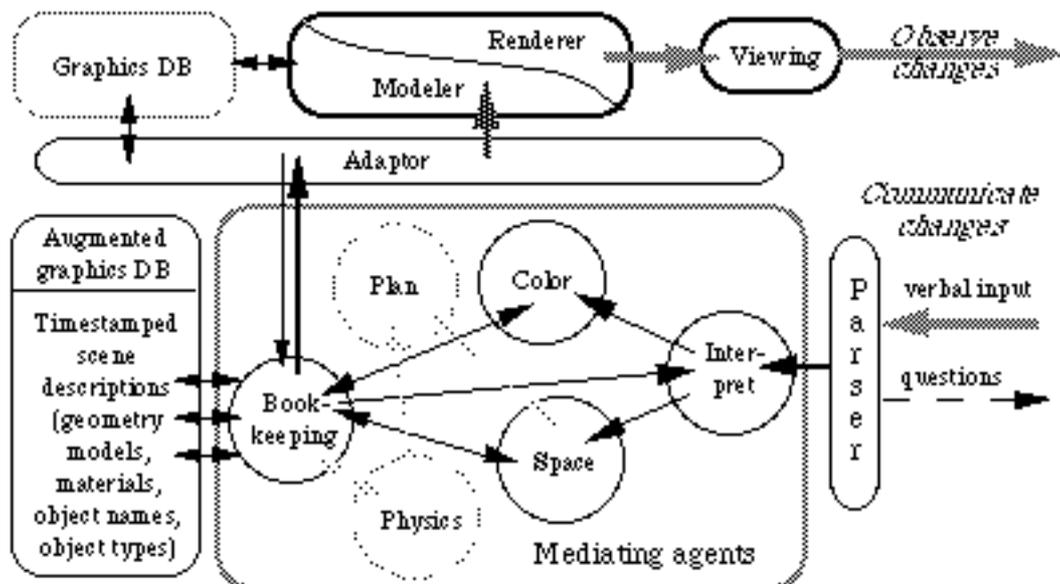


Fig. 7. Evolving architecture of the VIENA system: A multi-agent system mediates qualitative verbal instructions by translating them to quantitative commands that are used to update the visualization scene model (further explanation in text).

The core system is designed to be portable, hence a special adaptor unit converts data back and forth between the mediating agents and the loosely coupled graphics system. The adaptor also establishes a ‘pipeline’ to the modeler/renderer unit so that modified scene data can be visualized instantaneously. The graphics database is mirrored in the mediating system in an augmented graphics database which is local to a Bookkeeping agent. All data about the scene can only be modified via the Bookkeeping agent. Besides the current scene description, the augmented database holds information about previous scenes to be exploited in situated communication.

A major part of our work has concentrated on the technical environment in which mediating agents can come to bear. We have experimental implementations for an Adaptor unit (specific to the SOFTIMAGE modeler/renderer), a Parser unit, and an augmented graphics database (*cf.* Fig. 7). Based on this evolving working environment, we have begun to specify and to implement agents for intelligent mediating. We have worked on two types of changes so far, spatial and material, which are dealt with by the “Space” and the “Color” agent.

Adaptor

The Adaptor must translate data from SOFTIMAGE’s database to an augmented database and filter modified data to the renderer. In our experimental implementation we manipulate 3D scene objects by way of channels in the SOFTIMAGE (version 2.6.2) motion environment. At present, there are 1000 channels available where each channel is mapped exactly onto one pre-defined feature. In this way, about 80 objects can be controlled. The solution with SOFTIMAGE’s channel driver interface is limited in that we must prepare for each object which is to be controlled interactively. For this reason, we have also started to use the custom script language provided with the SOFTIMAGE actor environment. Custom scripts provide another means of interfacing user-defined functions with SOFTIMAGE. Using the Developer Kit, all types of scene data can be read and modified by way of appropriate script commands.

Augmented database/Bookkeeper

The modeler/renderer has its own data structures in a graphics database which holds information necessary for visualization. These data structures need to be augmented to be suitable for intelligent mediation. For instance, ‘non-visualizable’ (type) information is added that enables the system to distinguish between movable and non-movable scene objects, or to make reference to intrinsic perceptive features of scene objects. Also, previous features of a changed scene model are kept for evaluating elliptic discourse (“a little more”). To support such situated communication, data structures in the augmented database accommodate the scene history where each scene is tagged with a timestamp. Only the Bookkeeping agent, acting as a “gate keeper,” is authorized to access and modify the augmented database (*cf.* Figure 7). When piping a new scene for visualization, the augmented graphics database is updated accordingly. When a session is completed and quit, the current scene description is transferred to the SOFTIMAGE graphics database while the augmented database is abandoned.

Parser

While the Adaptor unit constitutes an interface between the agent system and the visualization, the parser unit interfaces between verbal input and the mediating agent system. In our setting we think it adequate that user input is kept to the minimum significant information. Thus we are not trying to process very complex sentential

structures. However, the system should be able to process discourse that negotiates the semantics of instructions. That is, if the effect of an instruction does not satisfy the user's expectations, corrective statements need to be analyzed based on previous discourse. Our parser (PARSY) translates simple qualitative English instructions to structured (deep-level) representations which output to an Interpret agent. The structured representation PARSY constructs for an instruction as "move the table to the right") contains an operator ("move"), an objectname ("table"), a degree ("") as well as a location ("right") or color specification. PARSY is also able to resolve simple ellipses such as "a little less". In Figure 8, a sample discourse and generated parser outputs are shown.

```

saturn%parsy
PARSY> move the table to the right
      move(obj=table, deg=, loc=right)
PARSY> a little less
      modify(obj=table, deg=little, loc=negativ)
PARSY> make the chair red
      make(obj=chair, deg=, col=red)
PARSY> a little more
      modify(obj=chair, deg=little, col=positiv)
PARSY> move the chair
      You must enter a direction.
PARSY> left
      move(obj=chair, deg=, loc=left)
PARSY>

```

Fig. 8. Input and internal output (bold face) of the parser PARSY

Within a session discourse, an elliptic phrase ("a little less") will be represented as a modification of the previous instruction. (A modify-command will make agents compare the most current scene model and the previous one by way of time-stamped entries in the augmented graphics database; see below.) When analyzing an incomplete instruction, PARSY may prompt the user for further input (*cf.* Figure 8). Again, consecutive input is processed in discourse.

Interpret agent

The structured representation of a verbal instruction is further elaborated by an Interpret agent which transmits commands to specialized agents for evaluating semantics. If an instruction refers to a previous one (e.g., "a little less" following "move the table to the right"), the system must exploit the session history to compute the semantics. To this end, the Interpret agent cooperates with the Book-

keeping agent to access the augmented database so the situated communication can successfully arrive at offering a new resulting scene. At present, the Interpret agent serves as a router which decides to which following agent (Space or Color) an instruction is issued. We take into account that certain instructions can be posted to several agents, e.g., “show the chair better” could be taken on by both the Color and the Space agent and result in an enhancement of the color of the chair, as well as in a move of the camera to change the aspect from which the scene is viewed.

Space agent

The Space agent has the following responsibilities: (1) to identify mentioned objects and (2) to determine where and how an object will be moved – in relation to other objects and avoiding collisions – in order to satisfy a user input. In the simple cases we have considered so far, only the target scene description is transferred for visualization. If a more complex scene modification is to take place, e.g., rearranging a collection of objects, a Plan agent – not yet realized – could become active and cooperate with other agents to generate a series of images.

The Space agent takes structured outputs from Parser, resembling qualitative instructions, and processes them to quantitative, absolute geometry data changes (i.e., translations in appropriate coordinates). The system is able to take into account all situational data of a current scene, including previous discourse. Thus, ambiguity in instructions can be greatly reduced. If a system response does not meet the user’s expectations, a follow-up instruction can build on the previous response.

To process a representation like **move(obj=table, deg=, loc=right)**, the Bookkeeping agent determines first which object named “table” is addressed by the instruction, and then reads the geometry data of this object to the Space agent. If more than one object is named “table”, the “table” object with the most current time stamp is selected, or further input is requested. The Space agent finds out which location is “right” in the current situation, how far “right” can range, and which moderate degree of a “right” move is offered in response to the command. More precisely, the space agent

- decides which is the current reference frame
- takes account of objects in the target area to avoid collisions
- calculates an according translation vector

The translation vector also depends on the size of the object to be moved (that is, a table undergoes a larger move than a bowl). The Adaptor transfers the change of “table’s” geometry data to the renderer, and the Bookkeeper updates the augmented database accordingly. To deal with a “modify” instruction, Bookkeeper accesses according entries in the augmented database and finds out the last change. A new change is inferred based on the last recorded change and the current change command.

Further work

Other agents will incrementally enhance the system’s ability for mediating verbal instructions. We have started to work on a Color agent which acts in a similar

fashion as the Space agent: a qualitative instruction is processed to quantitative material data changes (e.g., changings of rgb-values of objects or lights). We also consider conceptualizing subagents of an agent to take on special tasks. For instance, we work on a subagent for “Space” to determine which objects are *currently* on an object to be moved so they move with it (e.g., a bowl on a table). The general idea is that each agent is just “smart” enough to meet its special responsibility. By this, modular system development is supported.

6 Summary and Outlook

In this paper we have described how interaction modalities in the design and exploration of 3D computer graphics can be enhanced by “intelligent” (symbolic) communication. In the VIENA project, we use an example application of interior design to explore how verbally communicated changes of the scene can be put in effect by a mediating system. Our conceptualization of the mediating system involves a number of specialized subsystems – agents – which cooperate with the designer by exploiting internal scene descriptions.

An important part in this mediation is that agents transfer qualitative descriptions of scene changes to quantitative internal commands which can be processed by the graphics system. We also provide ways to deal with elliptic discourse (i.e., communications which refer back to previous ones) to make verbal interaction more natural. Finally, “situated” communication makes it possible for the user to issue commands from changing perspectives and by incorporating egocentric reference frames (like “left/right”).

Besides using agent ideas as helpful metaphors for human-computer-interaction, we have also found them useful for conceiving the architecture of the interface software in a modular way, and independent of a particular programming environment. That is, special-purpose machinery can be included in the designer’s workplace as necessary.

Interactivity is an important asset to support a creative process in graphics design. The so-far available means of interaction, e.g., the data glove, are but one way to manipulate the arrangement of a scene. Interaction modalities are further enhanced when designers can use language and symbolic qualities as input. On the other hand, our work on intelligent mediation is not restricted to solely verbal interaction; it seems also relevant for direct (gestural) manipulation. For instance, when a table is moved by grasping it with a data glove, the things on it could move with it by agent mediation as indicated above.

To conclude, we think that, in general, techniques from artificial intelligence can provide a more comfortable communication link between humans and multimedia. Eventually, we see the use of voice input and gestures as parallel input modalities. While we do not have equipment to exploit the usefulness of our work in a more immersive environment (e.g., with head-mounted display or a boom), we think it can be incorporated in such a setting at a later time.

References

- Adler, M., Durfee, E., Huhns, M., Punch, W., and Simoudis, E. (1992). AAAI workshop on cooperation among heterogeneous intelligent agents. *AI magazine* 13(2), 39-42.
- Badler, N.I., Webber, B.L., Kalita, J., and Esakov, J. (1991). Animations from Instructions. In N.I. Badler, B.A. Barsky and D. Zeltzer (eds.): *Making Them Move - Mechanics, Control, and Animation of Articulated Figures* (pp. 51-93). San Mateo (Cal.): Morgan Kaufmann.
- Bishop, G., Bricken, W., Brooks, F., Brown, M., Burbeck, C., Durlach, N., Ellis, S., Fuchs, H., Green, M., Lackner, J., McNeill, M., Moshell, M., Pausch, R., Robinett, W., Srinivasan, M., Sutherland, I., Urban, D., and Wenzel, E. (1992). Research Directions in Virtual Environments: Report of an NSF Invitational Workshop (March 23-24, 1992, University of North Carolina at Chapel Hill). *Computer Graphics* 26(3), 153-177.
- Böhm, K., Hübner, W. and Väänänen, K. (1992). GIVEN: Gesture driven interactions in virtual environments: A toolkit approach to 3D interactions. *Proc. Conf. Interface to Real and Virtual Worlds* (Montpellier, France, March 1992).
- Brooks, F. (1987). Walkthrough – A Dynamic Graphics System for Simulating Virtual Buildings. *Proc. of ACM Workshop on 3D Graphics* (Chapel Hill, Oct. 1987), 9-21.
- Brooks, R.A. (1991). Intelligence without reason. *Proceedings IJCAI-91*, 569-595.
- Bryant, D.J. (1991). *Perceptual Characteristics of Mental Spatial Models* (Dissertation). Stanford University: Dept. of Psychology.
- Cao, Y. (1993). *Zur Darstellung und Verarbeitung von Wissen über Himmelsrichtungen – Geometrische und Kognitionswissenschaftliche Aspekte* (Dissertation, University of Hamburg/Dept. of Computer Science). Sankt Augustin: Infix (DISKI-33).
- Chapman, D. (1991). *Vision, Instruction, and Action*. Cambridge (MA): MIT Press.
- Friederici, A.D. (1989). Raumreferenz unter extremen perzeptuellen Bedingungen: Perzeption, Repräsentation und sprachliche Abbildung. In Ch. Habel, M. Herweg and K. Rehkämper (eds.): *Raumkonzepte in Verstehensprozessen*. Tübingen: Niemeyer.
- Krüger, W. (1993). Virtual Reality – Anwendungen in Wissenschaft, Technik und Medizin. *it – Sonderheft "Multi Media"*, April 1993.
- Lang, E. (1989). Objektschemata und räumliche Konfiguration. In Ch. Habel, M. Herweg and K. Rehkämper (Hrsg.): *Raumkonzepte in Verstehensprozessen* (pp. 150-173). Tübingen: Niemeyer.
- Laurel, B. (1990). Interface agents: Metaphors with character. In B. Laurel (ed.) (1990): *The art of human-computer interface design*. Addison-Wesley.
- Levelt, W. (1986). Zur sprachlichen Abbildung des Raumes: Deiktische und intrinsische Perspektive. In H.-G. Bosshardt (Hrsg.): *Perspektiven auf Sprache* (S. 187-211). Berlin: DeGruyter.
- Maes, P. and Kozierok, R. (1993). Learning interface agents. Proceedings of the Eleventh National Conference on Artificial Intelligence (AAAI-93) (pp. 459-465). Menlo Park: AAAI Press/The MIT Press.
- Marcus, A. (1993). Future directions in advanced user interface design. In Thalmann, N.M. and Thalmann, D. (eds.): *Communicating with virtual worlds* (pp. 2-13). New York: Springer-Verlag.
- Meinkoehn, J. and Knoll, A. (1993). Recent results in organization and performance evaluation for large sensor networks. *Int. Conf. on Intelligent Autonomous Systems (IAS-3)*, Pittsburgh.
- Minsky, M.L. (1986). *The society of mind*. New York: Simon and Schuster, Inc.
- Steiner, D., Mahling, D., and Haugeneder, H. (1990). Human Computer Cooperative Work. In M. Huhns (ed.). *Proc. of the 10th International Workshop on Distributed Artificial Intelligence*. MCC Technical Report No. ACT-AI-355-90.
- Newell, A. (1981). The knowledge level (Presidential address, AAAI-80). *AI magazine* 2(2), 1-20.
- Zeltzer, D. (1991). Task-level graphical simulation: Abstraction, representation, and control. In N.I. Badler, B.A. Barsky, and D. Zeltzer (eds.): *Making Them Move - Mechanics, Control, and Animation of Articulated Figures* (pp.3-33). San Mateo (Cal.): Morgan Kaufmann.