Sonderforschungsbereich 378
Ressourcenadaptive kognitive Prozesse

KI-Labor am Lehrstuhl für Informatik IV

Leitung: Prof. Dr. W. Wahlster

REAL

Universität des Saarlandes
FB 14 Informatik IV
Postfach 151150
D-66041 Saarbrücken
Germany

Tel. 0681 / 302-2363

Bericht Nr. 132

Natural Language Access to Intelligent Robots: Explaining Automatic Error Recovery

Thomas Längle, Tim C. Lüth, Eva Stopp, Gerd Herzog

Oktober 1996

Sonderforschungsbereich 378
Ressourcenadaptive
Kognitive Prozesse

ISSN 0944-7814

132

Natural Language Access to Intelligent Robots: Explaining Automatic Error Recovery

Thomas Längle, Tim C. Lüth, Eva Stopp,* Gerd Herzog*

Institute for Real-Time Computer Systems and Robotics
University of Karlsruhe
D-76128 Karlsruhe, Germany

FB 14 Informatik
Universität des Saarlandes
D-66041 Saarbrücken, Germany

Abstract

The increasing intelligence and autonomy of modern robot systems requires new and powerful man-machine-interfaces. For example, a robot's capability to autonomously recover from error situations corresponds with dynamic adjustments of robot plans during execution. This makes it difficult for an operator to predict and understand the behaviour of the machine. Explanations and descriptions of why and how a certain plan has been changed would impose the acceptance of robots. Descriptions are even more important in error situations which can not be handled autonomously. In this context, natural language is an effective tool of using robots in a more flexible manner. In this article, the joint efforts of the University of Karlsruhe and the University of Saarland to provide a natural language explanation for the error recovery of the autonomous, mobile, two-arm robot, *KAMRO* are reported.

In: A.M. Ramsay (ed.): Artificial Intelligence: Methodology, Systems, Applications, Proc. of AIMSA'96, Sozopol, Bulgaria, September 1996, Amsterdam: IOS Press, 259–267.

1 Introduction

As advanced robot systems steadily gain higher intelligence and greater autonomy, the requirements for the design of flexible interfaces to control these systems will increase. This is for instance the case, according to [Rembold et al., 1993], for the future use of intelligent robots in manufacturing or as service robots for different applications. Therefore, natural language as a communication medium for humans is an efficient means of making a technical system easier accessible to its users [Wahlster, 1989]. A practical advantage of natural language access is the possibility of conveying information in varying degrees of condensation and of communicating on different levels of abstraction in an application-specific way.

In this article, the joint efforts of the University of Karlsruhe and the University of Saarland to provide natural language access to the autonomous, mobile, two-arm robot, *KAMRO* (**K**arlsruhe **A**utonomous **M**obile **Ro**bot), which is being developed at IPR, are reported. In the last two years, research has focused on the topic of task specification by use of natural language: operations and tasks which are to be performed by the robot can be specified at different levels. The most abstract specification would just provide a qualitative description of the desired state, i.e., the final positions of the different assembly parts. During task execution, the capability of recovering from error situations leads to a dynamic adjustment of the robot's plans. This feature makes it more difficult for the operator to predict and understand the behaviour of the robot. Explanations and descriptions of why and how a plan was changed would increase the cooperativeness of an intelligent robot. Thus, this article will focus on the extension of the implemented interface in allowing the use of natural language explanations to describe the intelligent robot's error recovery.

2 Related Work

The potential benefit of man-machine interaction in natural language was recognised quite early, as shown, for example, by [Sondheimer, 1974]. However, up to now, there have only been a few attempts towards natural language access for robot systems. The well known mobile robot *SHAKEY*, as described in [Nilsson, 1984], is able to understand simple commands given in natural language. [Sato and Hirai, 1987] concentrate on the language-aided instruction of a remote robot system. In their approach, specific words may be used to simplify the specification of teleoperational functions. [Torrance, 1994] introduced a navigating, indoor, office-based, mobile robot, capable of understanding basic commands and answering simple queries related to the navigation task. [Moratz et al., 1995] aim at the natural language instruction of an assembly robot and present a framework for the integration of speech understanding and visual perception. The connection between perceptual information derived from sensory data on the one hand and verbal expressions on the other hand constitutes a prominent issue for natural language access to robot systems. First results concerning the integration of vision and language processing have already been achieved in the context of image sequence analysis, [Herzog and Wazinski, 1994, Neumann, 1989, Wahlster et al., 1983]. In addition, the approaches of [Badler et al., 1991, Chapman, 1991, Vere and Bickmore, 1990, Wachsmuth and Cao, 1994] are relevant for verbal man-machine interaction, since they consider natural language control of autonomous agents within simulated 2D or 3D environments.

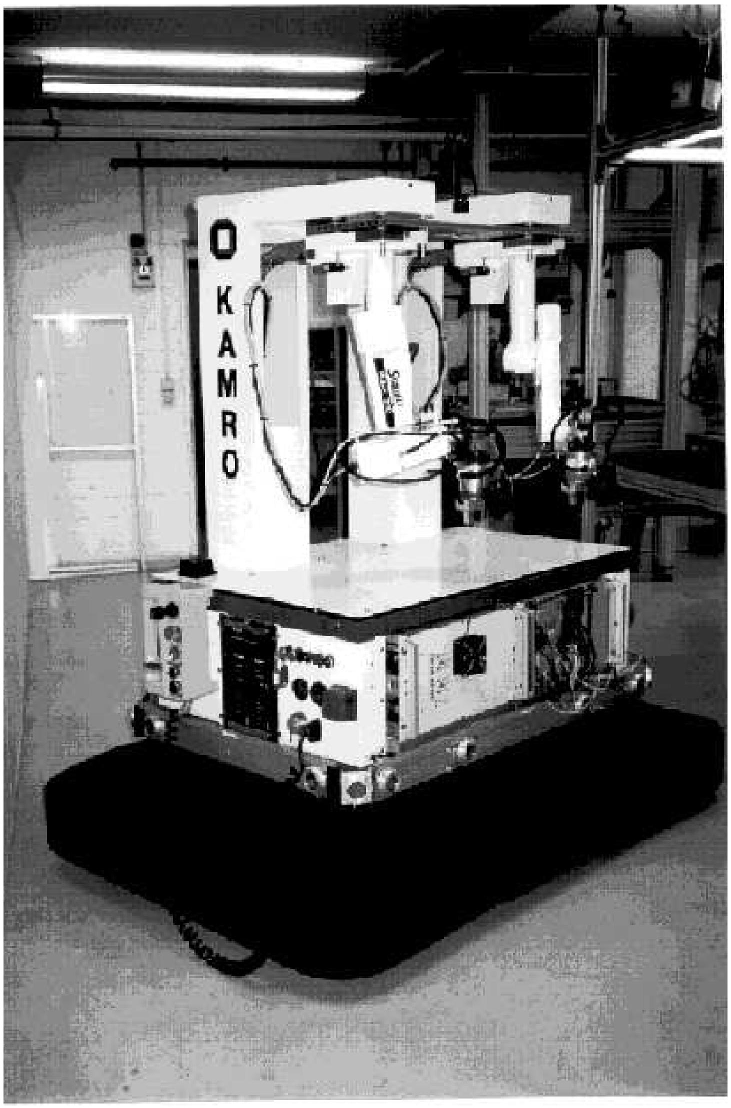

Figure 1: The mobile robot *KAMRO*

3 Interaction with the Robot

In the *KAMRO* project, an autonomous mobile robot (Fig. 1) designed for assembly tasks is being developed which also has the capability of recovering from error situations, [Lüth and Rembold, 1994]. The autonomous, mobile robot *KAMRO* is a two-arm robot system that consists of a mobile platform with an omnidirectional drive system, two Puma 260 manipulators, and different sensors for navigation, docking and manipulation.

KAMRO is capable of performing assembly tasks (Fig. 2) autonomously. The tasks, or robot operations, can be described on different levels at abstraction: assembly precedence graphs, implicit robot-independent elemen-tary operations (pick, place), and explicit robot- dependent elementary operations (grasp, transfer, fine motion, join, exchange, regrasp, set gripper, detach).

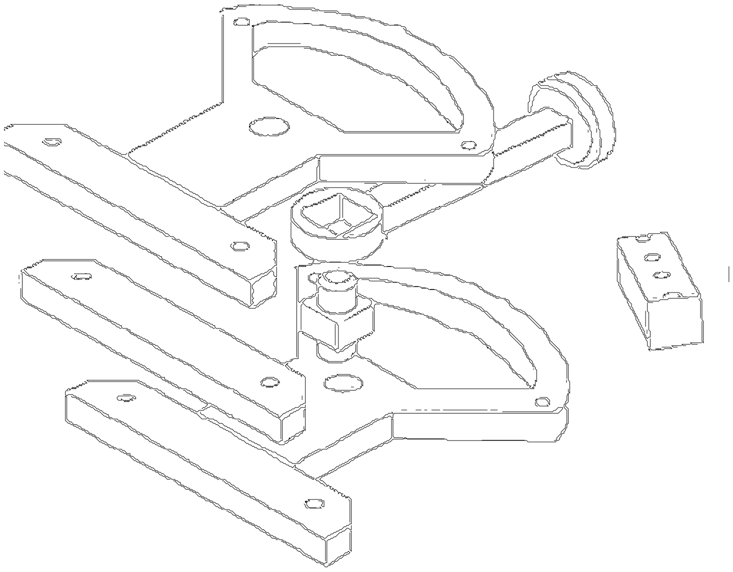

Figure 2: The Cranfield Assembly Benchmark

A given complex task is transformed by the control architecture (Fig. 3) from the assembly precedence graph level to the explicit elementary operation level. The generation of suitable sequences of elementary operations depends upon the position and orientation of the assembly parts on the worktable, while execution is controlled by the real-time robot control system. Status and sensor data which is given back to the planning-system enables KAMRO to control the execution of the plan and to correct it, if necessary. In the context of natural language access to KAMRO four main situations of man-machine interaction can be identified [Lüth et al., 1994]:

- *Task specification:* Operations and tasks can be given at different levels of abstraction.

- *Explanation of error recovery:* The capability of the robot to recover from error situations can cause comprehension problems for the user. Explanations of why and how plans are changed increases cooperativeness.

- *Execution monitoring:* Assembly tasks can sometimes be performed in different sequences and the autonomous robot can decide for itself the final sequence of operations. Therefore, it will be clearer for the user if he is informed about the actions of the robot.

- *Updating and describing the environment representation:* On the one hand, the robot can give a verbal description of the scene. On the other hand, the user can provide information about parts of the scene which are outside the visual field of the robot.

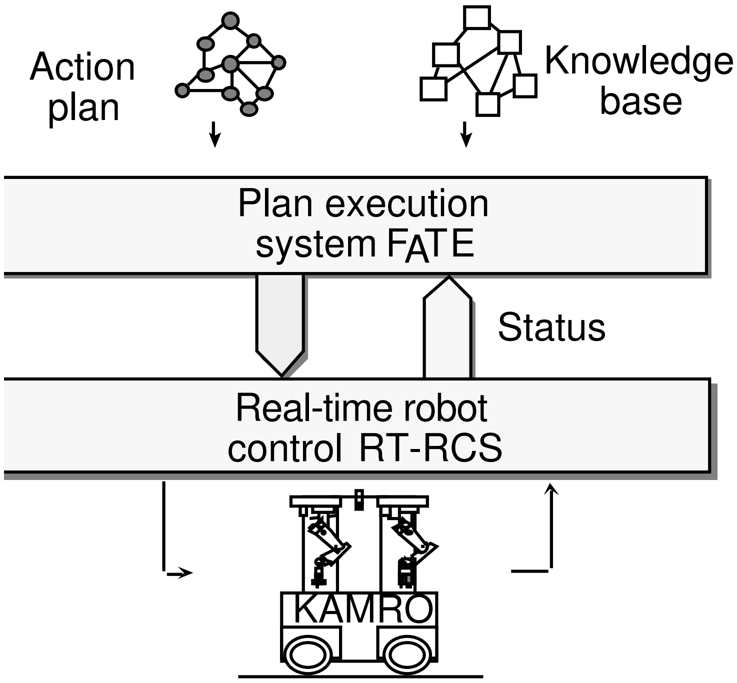

Figure 3: Structure of the *KAMRO* System

4 Error Recovery

In this section, the error recovering component of the KAMRO robot is described, [Dumm, 1991]. In an assembly domain (see Fig. 2), two main classes of errors can be distinguished:

1. Errors that can be avoided by the system's fault tolerances, [Trevelyan and Nelson, 1987].

2. Errors which occur by inconsistency of the world model and which need to be recovered, [Srinivas, 1977].

To be more specific, three different situations that require fault-tolerant behaviour can be distinguished:

1a. Faulty system components, for example, a camera system which does not work any longer because a computer controlled light source is damaged.

1b. Errors that occur due to uncertain information, for example, inconsistency of the world model if an assembly part is not at its predefined position (Fig. 4a).

1c. Reduced reliability of system components, for example, if their accuracy is no longer satisfactory.

In order to realize these tasks the *KANTRA* system (**KA**MRO **N**atural Language **Tra**nslator) was developed based on work done in the *VITRA* project. So far, the focus was on the role of spatial relations in such a system [Stopp et al., 1994], especially for natural language task specification concerning implicit elementary operations [Längle and Lüth, 1995, Längle et al., 1995].

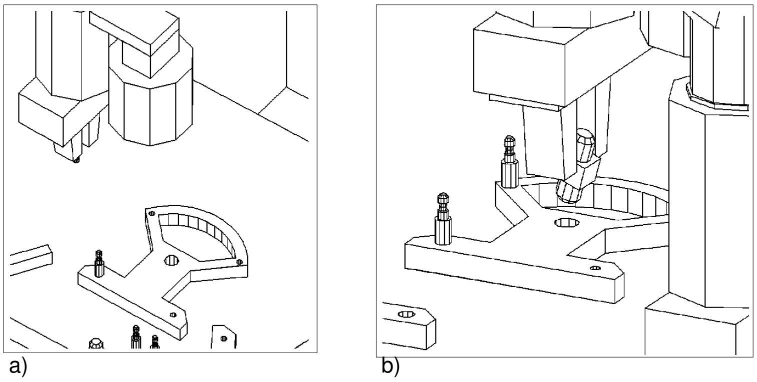

Figure 4: a) Error that is caused by uncertain position information, b) Error that must be recovered

By the use of error recovery modules, it should be possible to guarantee correct system behaviour after an error situation has already occurred. Four types of errors are distinguished:

2a. Information error (uncertainties of the world model, for example, a slipped object in the gripper, see Fig. 4b).

2b. Operational error (for example, a collision between an object and a manipulator).

2c. Constraint error (for example, an immovable object that has to be gripped).

2d. Precondition error (not correctly executed preceding operations).

The fault-tolerant behaviour of the robot is guaranteed by the use of a distributed control architecture, [Längle and Lüth, 1995, Rembold et al., 1993]. In this way, it is possible to prevent error situations in advance. Here, the system is equipped with several sensor systems that can be used to perceive important information in order to prevent error situations. For example, the hand camera is used just before executing a gripping or joining operation to correct the position of the base object in the world model. Through this, position uncertainty can be reduced.

All operations that a robot can perform are based on operational models. Error recovery is the ability to guarantee correct system behaviour in situations when the world state does not match the world model because it is inconsistent with the operational model. Thus, error recovery is the ability to transfer the system state into a consistent one in situations which have not been taken into account in advance. This has to be done if, for example, a manipulator has gripped a wrong object, or if a hand camera can not find the base object before execution of a join operation. Furthermore, it is possible that a gripped object may slightly slip in the gripper.

It is difficult to guarantee error recovery in unexpected situations because all online occurring problems as well as their solutions must be taken into account in advance to teach the robot system correct behaviour. To overcome these problems, the system uses model-based error detection, error diagnosis and error recovery.

In the case of a camera system (hand camera, overhead camera), an error occurs if the camera system has to calculate the position of an object, and the object can not be found. These errors are easy to detect.

In the case of a manipulator, error detection is more difficult because there are many error possibilities. Here, the manipulator has to evaluate sensory information before, during and after an operation is executed. In that way, the manipulator is able to recognize a wrong position, or that a gripped object is not in the gripper any longer. While performing an operation, execution time is also used to detect an error.

After an error is detected, it is the robot's task to explain the error situation and to start recovery operations. Therefore, the error is divided up into the already described error classes: information error, operational error, constraint error and precondition error. For each error class, the corresponding error probability is calculated. In this way, the system has the ability to investigate errors in a comprehensive way.

For each operation class, a different error recovery routine is used to solve the problem. Thus, in situations in which a manipulator is, as an example, not able to perform an assembly task because it has gripped a wrong object, it has the ability to verify the system state by use of the camera system once the manipulator has placed the object onto the table. After the error diagnosis, the robot is able to give a presumed cause of the error and the explicit elementary operation which is responsible for it.

All these *diag* information is available to the dialog component in an internal representation format:

```
diag=(<IEO>, <error>, <cause>, <EEO>, <goal>)
```

<IEO> represents the implicit robot operation where the error occurred, <error> is a representation of the error itself, <cause> is the presumed cause of the error, <EEO> the corresponding explicit elementary operation of the error, and <goal> the high-level goal of the error recovery plan of the presumed cause.

Example

IEO:	place shaft
error:	not (position (right-object))
EEO:	join (place)

In this example, the robot intends to join an assembly part, the shaft, to the baseplate. During the joining operation, the interpretation of the force-torque-sensor information differs too much from the sensor model. In this case, different causes are possible:

- the robot has gripped the wrong object (precondition error),
- the robot has gripped the right object but in a wrong way (precondition error),
- the object has slipped in the gripper (information error),
- the base object is not at the right place (operational error), or
- the hole of the baseplate is too small or the shaft is too big (constraint error).

Constraint errors cannot be recovered. All other error causes are ordered referring to their occuring probability. In this example, the information error has the highest probability:

cause: not (position (right-object))

This error can be removed by the corresponding error recovery procedure: verifying and updating the world model.

goal: verify-world-model

Thus, the object is placed by the robot on the table, a new camera snapshot is taken, and the object is regrasped by the manipulator. The error diagnosis can be used in a recursive manner if, for example, an error arises during the error recovery.

5 Explaining the Error Recovery

In planning natural language explanations for the mobile robot's error recovery during a single, planning step, it is necessary to find out what is important for user comprehension. This information builds the basis of the verbalization. Here, four different levels of information which can be verbalized in an explanation can be identified:

- error: current robot operation and assumed error,
- cause: cause and current robot operation,
- solution: goal of the error recovery action,
- recovery plan: the different steps of the error recovery action.

Since the knowledge of the user varies, these levels can be used for the generation of utterances in more or in less detail. Perhaps a changing initiative, in this context, makes a dialog-based approach necessary.

In order to design a natural language component for KAMRO which is readily integrated into the system, it is necessary to use knowledge that the robot already has internally. This knowledge can be related to the described levels of information identified for the generation of an explanation:

- error: implicit robot operation where the error was realized (IEO), assumed error
- cause: cause of the error, explicit robot operation which caused the error (EEO)
- solution: assumed solution goal
- recovery plan: recovery plan

Realizing a dialog-based approach, it is important to consider where the user can intervene. As a first strategy, the system will begin with the error and cause levels and wait for the user's intervention before giving more information. Since the interface works in a restricted domain with a limited number of recognisable errors with corresponding error recovery plans, they can be related to text patterns as suggested in [McKeown, 1985]. For the first three levels, suitable text frames look like this:

1. While trying to <implicit robot operation where the error was realized (IEO)> it was recognised that <assumed error>.
2. When <explicit robot operation which caused the error (EEO)> <cause of the error>.
3. In order to recover it can be tried to <assumed solution goal>.

<EEO>, <IEO>, <error>, <cause> and <goal> correspond to verbalizations of the actual instances delivered by *KAMRO*. A conceptual lexicon here has to relate these instances to a text plan which is the basis for the verbalization. For example, a robot operation corresponds to a verb frame with some information about the verb to be used and slots to be filled in order to get all information needed, e.g.:

place shaft:	ACTION:	place
	AGENT:	manipulator
	PATIENT:	shaft

Not every slot needs to be verbalized together with the text frame. In contrast to the patient, the agent slot is known but is in most cases redundant because for each IEO and EEO the agent is one of the manipulators. So, the action and the patient is used for the verbalization. Another problem is the use of referential expressions for the objects, i.e., the selection of definite or indefinite determiners and the use of pronouns. However, these are problems which have to be solved together with the general problem of choosing the suitable text frame for the actual context by including a user model and a dialog history. Since, in the example, the robot performes a place-operation and the object should be known by the user from the preceeding pick-operation, the noun with its definite determiner as a referential expression for the object `shaft' is used.

The verbalization of the goal-instance is done in a similar way since it also corresponds to an action, sometimes with, sometimes without a parameter depending on the goal which has to be reached. Error- and cause-instances correspond to states of the world. At the end of processing the following explanation is verbalized for the example:

1. While trying to place the shaft it was recognised that the object has probably slipped in the gripper.

2. When joining the shaft to the baseplate, the force-torque sensor information differed from the sensor model.

3. In order to recover, it is tried to update the world model and grasp the object again.

The verbalization of the fourth level differs from the other levels because it is not possible to define one homogeneous pattern for all plans. In contrast, the interface has to verbalize them by concatenating verbalizations of the planning steps. So, although the procedure is similar, a higher flexibility can be guaranteed since the interface has the possibility to vary the text for every plan instance. In order to make the resulting text less redundant it is useful to summarize commands if possible. For the plan described in section four the explanation follows a pattern like this:

First, <task1&2>, then <task3> in order to <task4>.

First, the manipulators are moved to their standard position, then the camera system is activated in order to decide if the operations were based on the wrong world model. In the example, <task1> and <task2> are summarized as <task1&2> because they concern the same operation merely with different instances of the same object type. The verbalization of the tasks follows a similar scheme as for the verbalization of the robot commands described above.

6 Evaluation

KAMRO constitutes an advanced autonomous mobile assembly robot with a natural language interface for flexible man-machine interaction. An outstanding feature of the intelligent system is its capability to recover automatically from error situations during task execution. The dynamic adjustment of assembly plans, however, makes it more difficult for the operator to predict and understand the behaviour of the robot. In such situations, human-robot interaction in natural language increases the cooperativeness of the autonomous system. In order to generate verbal descriptions which explain the robots behaviour, it is necessary to exploit the internal knowledge structures underlying the error recovery mechanism of the intelligent system. Accordingly, the natural language descriptions are structured into several consecutive parts providing different levels of detail. In combination with the dialog-based approach favoured in the system the explanation text may be tailored to a user's actual needs. Thus, user adaptive communicative behaviour is achieved.

Currently, the system is restricted to the generation of natural language explanations for error recovery within a single high-level action during plan execution. For the purpose of execution monitoring the global assembly plan also has to be taken into account. This allows even more sophisticated explanations and descriptions in natural language. Such an extension, however, will require refinements of the dialog strategies in order to guarantee a flexible but nonetheless coherent discourse.

7 Conclusion

The aim of efforts in combining an autonomous mobile robot system with a natural language access system is the development of a more cooperative interface which facilitates man-machine interaction. In this contribution, there was a focus on the problem of generating natural language explanations for automatic error recovery, as may be carried out by the intelligent robot *KAMRO* while performing an assembly task.

It is argued that the user can not always anticipate the complex autonomous behaviour of the intelligent system, and therefore the need for providing adequate verbal descriptions arises in order to support the understanding and the supervision of the robot's actions. The framework relies on a flexible dialog-based approach and exploits the internal knowledge sources of the autonomous system for the generation of comprehensive textual descriptions. The results obtained so far form a promising basis for further extending the communicative capabilities of the *KAMRO* system.

8 Acknowledgement

This research was performed at the Institute for Real-Time Computer Systems and Robotics (IPR), Prof. Dr.-Ing. U. Rembold and Prof. Dr.-Ing. R. Dillmann, Faculty for Computer Science, University of Karlsruhe, and at the Department of Computer Science, University of the Saarland, Saarbrücken, Prof. Dr. W. Wahlster. This project is funded by the German Research Foundation (DFG) within the nationally based

research projects on artificial intelligence (SFB 314, until 1995) and on resource-adaptive cognitive processes (SFB 378, since 1996).

References

[Badler et al., 1991] **N. I. Badler, B. L. Webber, J. Kalita, and J. Esakov**. Animation from Instructions. In: N. I. Badler, B. A. Barsky, and D. Zeltzer (eds.), *Making Them Move: Mechanics, Control, and Animation of Articulaited Figures*, pp. 51–93. Morgan Kaufmann, San Mateo, CA, 1991.

[Chapman, 1991] **D. Chapman**. *Vision, Instruction, and Action*. MIT Press, Cambridge, MA, 1991.

[Dumm, 1991] **M. Dumm**. *Automatische Behandlung von Ausnahmesituationen bei Montagevorgängen*. Master's thesis, Fakultät für Informatik, Universität Karlsruhe, 1991.

[Herzog and Wazinski, 1994] **G. Herzog and P. Wazinski**. VIsual TRAnslator: Linking Perceptions and Natural Language Descriptions. *Artificial Intelligence Review*, 8(2), 1994.

[Längle and Lüth, 1995] **T. Längle and T. Lüth**. Decentralized Control of Distributed Intelligent Robots and Subsystems. In: *AIRTC Symposium on Artificial Intelligence in Real-Time Control*, pp. 447–452, Valencia, Spain. 1995.

[Längle et al., 1995] **T. Längle, T. Lüth, G. Herzog, E. Stopp, and G. Kamstrup**. KANTRA - A Natural Language Interface for Intelligent Robots. In: U. Rembold, R. Dillman, L. O. Hertzberger, and T. Kanade (eds.), *Proc. of the 4th International Conference on Intelligent Autonomous Systems*, pp. 357–364, Karlsruhe, Germany. 1995.

[Lüth and Rembold, 1994] **T. C. Lüth and U. Rembold**. Extensive Manipulation Capabilities and Reliable Behaviour at Autonomous Robot Assembly. In: *Proc. of IEEE Int. Conf. on Robotics and Automation*, San Diego, CA. 1994.

[Lüth et al., 1994] **T. C. Lüth, T. Längle, G. Herzog, E. Stopp, and U. Rembold**. KANTRA: Human-Machine Interaction for Intelligent Robots Using Natural Language. In: *3rd IEEE Int. Workshop on Robot and Human Communication, RO-MAN'94*, pp. 106–111, Nagoya, Japan. 1994.

[McKeown, 1985] **K. R. McKeown**. Discourse Strategies for Generating Natural-Language Text. *Artificial Intelligence*, 27(1):1–41, 1985.

[Moratz et al., 1995] **R. Moratz, H.-J-Eickmeyer, B. Hildebrandt, F. Kummert, G. Rickheit, and G. Sagerer**. Integrating Speech and Selective Visual Perception Using a Semantic Network. In: *AAAI Symposium on Computational Models for Integrating Language and Vision*, Cambridge, MA, USA. 1995.

[Neumann, 1989] **B. Neumann**. Natural Language Description of Time-Varying Scenes. In: D. L. Waltz (ed.), *Semantic Structures: Advances in Natural Language Processing*, pp. 167–207. Lawrence Erlbaum, Hillsdale, NJ, 1989.

[Nilsson, 1984] **N. J. Nilsson**. Shakey the Robot. Technical Note 323, Artificial Intelligence Center, SRI International, Menlo Park, CA, 1984.

[Rembold et al., 1993] **U. Rembold, T. C. Lüth, and A. Hoermann**. Advancement of Intelligent Machines. In: *ICAM JSME Int. Conf. on Advanced Mechatronics*, pp. 1–7, Tokyo, Japan. 1993.

[Sato and Hirai, 1987] **T. Sato and S. Hirai**. Language-Aided Robotic Teleoperation System (LARTS) for Advanced Teleoperation. *IEEE Journal on Robotics and Automation (RA)*, 3(5):476–480, 1987.

[Sondheimer, 1974] **N. K. Sondheimer**. English as a Command Language for Machines and the Semantics of "Left" and "Right". In: *Milwaukee Symposium on Automatic Control (and Autonomous Computing)*, Milwaukee, WI. 1974.

[Srinivas, 1977] **S. Srinivas**. *Error Recovery in Robot Systems*. Ph.D. thesis, California Institute of Technology, USA, 1977.

[Stopp et al., 1994] **E. Stopp, K.-P. Gapp, G. Herzog, T. Längle, and T. C. Lüth**. Utilizing Spatial Relations for Natural Language Access to an Autonomous Mobile Robot. In: B. Nebel and L. Dreschler-Fischer (eds.), *KI-94: Advances in Artificial Intelligence*, pp. 39–50. Springer, Berlin, Heidelberg, 1994.

[Torrance, 1994] **M. C. Torrance**. *Natural Communication with Robots*. Master's thesis, MIT, Department of Electrical Engineering and Computer Science, Cambridge, MA, 1994.

[Trevelyan and Nelson, 1987] **J. Trevelyan and M. Nelson**. Adaptive Robot Control Incorporating Automatic Error Recovery. In: *Proc. of the International Conference on Advanced Robotics*. 1987.

[Vere and Bickmore, 1990] **S. Vere and T. Bickmore**. A Basic Agent. *Computational Intelligence*, 6(1):41–60, 1990.

[Wachsmuth and Cao, 1994] **I. Wachsmuth and Y. Cao**. Interactive Graphics Design with Situated Agents. In: W. Strasser and F. Wahl (eds.), *Graphics and Robotics*, pp. 73–75. Springer, Berlin, Heidelberg, 1994.

[Wahlster, 1989] **W. Wahlster**. Natural Language Systems: Some Research Trends. In: H. Schnell and N. O. Bernsen (eds.), *Logic and Linguistics: Research Directions in Cognitive Science - European Perspectives, Vol. 2*, pp. 171–183. Lawrence Erlbaum, Hillsdale, NJ, 1989.

[Wahlster et al., 1983] **W. Wahlster, H. Marburger, A. Jameson, and S. Busemann**. Over-answering Yes-No Questions: Extended Responses in a NL Interface to a Vision System. In: *Proc. of the 8th IJCAI*, pp. 643–646, Karlsruhe, FRG. 1983.