Proceedings of the 2002 IEEE
International Conference on Robotics & Automation
Washington, DC • May 2002

Control Architecture and Experiment of A Situated Robot System for Interactive Assembly

Jianwei Zhang
zhang@techfak.uni-bielefeld.de
Faculty of Technology, University of Bielefeld,
33501 Bielefeld, Germany

Alois Knoll
knoll@informatik.tu-muenchen.de
Technical University of Munich,
81667 Munich, Germany

Abstract

We present the development of and experiment with a robot system showing cognitive capabilities of children of three to four years. We focus on two topics: assembly by two hands and understanding human instructions in natural language as a precondition for assembly systems being perceived by humans as "intelligent". A typical application of such a system is interactive assembly. A human communicator sharing a view of the assembly scenario with the robot instructs the latter by speaking to it in the same way that he would communicate with a child. His instructions can be under-specified, incomplete and/or context-dependent.

After introducing the general purpose of our project, we present the hardware and software components of our robots necessary for interactive assembly tasks. The control architecture of the robot system with two stationary robot arms is discussed. We then describe the functionalities of the instruction understanding, planning and execution levels. The implementations of a layered-learning methodology, memories and monitoring functions are briefly introduced. Finally, we outline a list of future research topics for extending our system.

1 Introduction

Human-beings interact with each other in a multimodal way. By reviewing the history of robotics, the modalities of human-robot interaction can be classified into three levels: *explicit* level, *implicit* level, and *inter-human like* level. With the enhancement of robot intelligence and advance of human perception, human-robot interaction can be developed naturally and *inter-human like*. A user can instruct a robot by using natural language (NL), gesture and gaze information in the way he communicates with a human partner. Technologies leading towards such a natural interaction with robots will contribute to the extension of robotic applications to all *human-in-the-loop* systems such as service robots, medical robots, entertainment robots, software robots, etc. In mechatronic applications, the "*machine intelligence quotient*" (MIQ) can be enhanced so that untrained persons can use such high functional devices easily. For building a robot system which understands human natural instructions, a robot control architecture which enables multimodal input, global memory access and fault monitoring becomes a central topic.

2 Some relevant work

One challenge of the research program for robotics is to automate the process of multisensor supported assembly by gradually enabling the robot and sensor system to carry out the individual steps in a more and more autonomous fashion. The typical hierarchical RCS architecture for realizing such systems was explained in details in [1]. However, a fully automatic assembly under diverse uncertain conditions can rarely be realized without any failure. Several projects on communicative agents realized with real robots have been reported, e.g. [8]. In the projects described in [2] and [10], natural language interfaces were used as the "front-end" of an autonomous robot. If constrained natural language is used to realise a limited number of robot operations, special steps can be taken, e.g. by only recongizing nouns in an instruction and listing the possible actions based on a pre-defined knowledge database [11]. In the SAIL project [10], level-based AA-learning combined with attention-selection and reinforcement signals was introduced to let a mobile robot learn to navigate and to recognize human faces and simple speech inputs. In [7], the main system architectures were compared, and an object-based approach was proposed to help manage the complexity of intelligent machine development. In the Cog project [3], the sensory and motor systems of a humanoid robot and the implemented active sensing and social behaviors were studied.

To overcome the limitations of this approach, the concept of the "Artificial Communicator" was developed, which we briefly outline in the sequel.

3 The Communicator Approach

If the nature of assembly tasks cannot be fully predicted, it becomes inevitable to decompose them into more elementary actions. Ideally, the actions specified are atomic in such a way that they always refer to only one step in the assembly of objects or aggregates, i.e. they refer to only one object that is to be assembled with another object or collection thereof (ag-

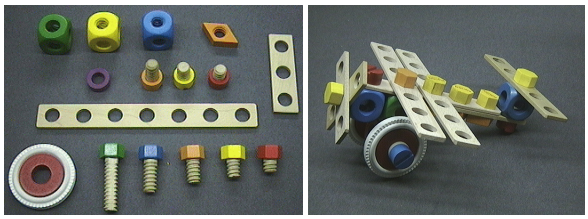
gregates). The entirety of a system that transforms suitable instructions into such actions is called an *artificial communicator* (*AC*). It consists of sensor subsystems, NL processing, cognitive integration and the robotic actors. From the instructor's point of view the AC should resemble a human communicator (*HC*) as closely as possible [6]. The AC must be seamlessly *integrated* into the handling/manipulation process. More importantly, it must be *situated*, which means that the situational context (i.e. the state of the AC and its environment) of a certain NL (and further modalities) input is always considered for its interpretation. The process of interpretation, in turn, may depend on the history of utterances up to a certain point in the conversation. It may be helpful, for example, to clearly state the goal of the assembly before proceeding with a description of the atomic actions. There are, however, situations in which such a "stepwise refinement" is counter-productive, e.g. if the final goal cannot be easily described. Studies based on observations of children performing assembly tasks have proven to be useful in developing possible interpretation control flows. From an engineering perspective, the two approaches can be likened to *open loop control* (Front-End Approach) and *closed loop control* (Incremental Approach) with the human instructor being part of the closed loop.

The research described in the following sections is embedded into a larger interdisciplinary research project aiming at the development of ACs for various purposes and involving scientists from the fields of computer linguistics, cognitive linguistics, computer science and electrical engineering.

4 The Situated Artificial Communicator

There is ample evidence that there exists a strong link between human motor skill and cognitive development (e.g. [5]). Our abilities of emulation, mental modeling and planning of motion are central to human intelligence [4] and, by the way, a precondition for anticipation, but they also critically depend on the experience we make with our own body dynamics as we plastically adapt our body's shape to the environment. As a basic scenario, the assembly procedure of a toy aircraft (constructed with "Baufix" parts, see Fig. 1) was selected. We have been developing a two-arm robotic system to model and realize human sensorimotor skills for performing assembly tasks and to facilitate human interaction with language and gestures. This robotic system serves as the major test-bed of the ongoing interdisciplinary research program of the project SFB[1] 360 "Situated Artificial Communicators" at the University of Bielefeld [13]. A number of parts must be recognized, manipulated and built together to construct the model aircraft. Within the framework of the SFB, in each of these steps, a human communicator instructs the robot, which implies that the interaction between them plays an important role in the whole process.

[1]Collaborative research unit funded by the Deutsche Forschungsgemeinschaft (DFG).



(a) The Baufix construction parts.

(b) The goal aggregate.

Figure 1: The assembly of a toy aircraft.

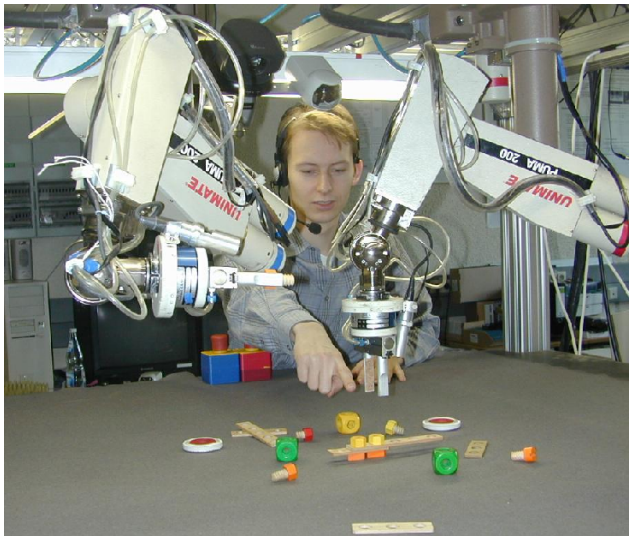


Figure 2: The two-arm multisensor robot system for dialogue-guided assembly.

The physical set-up of this system consists of the following components (Fig. 2):

(i) Two 6 d.o.f. PUMA-260 manipulators are installed overhead in a stationary assembly cell. On each wrist of the manipulator, a pneumatic jaw-gripper with integrated force/torque sensor and "self-viewing" hand-eye system (local sensors) is mounted.

(ii) Two cameras with controllable zoom, auto-focus and aperture provide the main vision function. Their tasks are to build 2D/3D world models, to supervise gross motion of the robot as well as to trace the hand and viewing direction of the human instructor.

(iii) A microphone and loudspeakers are connected with a standard voice recognition system, *IBM ViaVoice*, to recognize the human speech instructions and to synthesize the generated speech output.

5 Control Architecture

As the backbone of an intelligent system, the control architecture of a complex technical system describes the functionality of individual modules and the interplay between them. We developed an interactive hierarchical architecture according to Fig. 3. A HC is closely involved in the whole assembly process.

5.1 High-level functions

The system and the HC interact through natural speech and with hand gestures. First, an instruction is spoken to the robot system and recognized with the *ViaVoice* speech engine. In the current system, *ViaVoice* recognizes only sentences, which the grammar we developed allows. In practice, hundreds of grammar rules can be used. If the recognition succeeds, the results are forwarded to the speech recognition/understanding module.

By their very nature, human instructions are situated, ambiguous, and frequently incomplete. In most cases, however, the semantic analysis of such utterances will result in sensible operations. An example is the command "*Grasp the left screw*". The system has to identify the operation (*grasp*), the object for this operation (*screw*), and the situated specification of the objects (*left*).

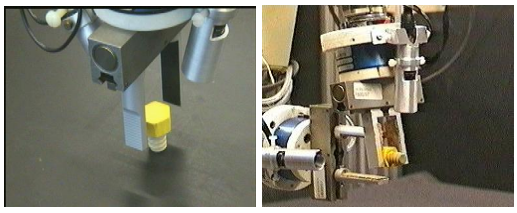
With the help of a hand gesture the operator can further disambiguate the object. The system may then use the geometric knowledge of the world to identify the right object. Other situated examples are: "*Insert in the hole above*", "*Screw the bar on the downside in the same way as on the upside*", "*Put that there*", "*Rotate slightly further to the right*", "*Do it again*", etc.

The output of the analysis is then verified to check if the intended operation can be carried out. If in doubt, the robot agent asks for further specifications or it has the right to pick an object by itself. Once the proper operation is determined, it is given to the *coordination* module on the next level. The final result on this level consists of an *Elementary Operation* (EO) and the objects to be manipulated with the manipulation-relevant information such as type, position/orientation, color, pose (standing, lying, etc).

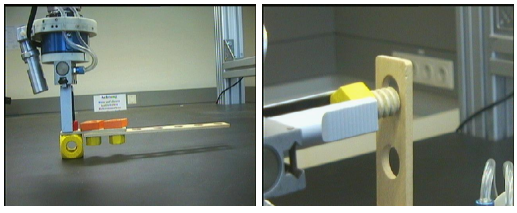
An EO is defined in this system as an operation which does not need any further action planning. Typical EOs are: *grasp*, *place*, *insert into*, *put on*, *screw*, *regrasp*, *alignment* (for an illustration see Fig. 4). The robustness of these operations mainly depends on the quality of the different skills.

5.2 Planning tasks

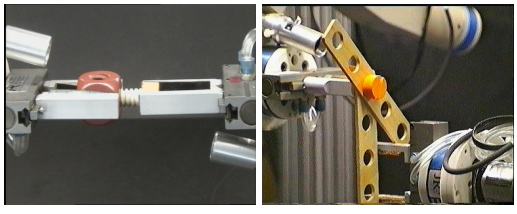
On the planning level, an assembly task of the toy aircraft, or of sub-aggregates, is decomposed into a sequence of EOs. The final decision about the motion sequence depends on the instructions of the human user as well as the generated plan. The *planning* module should not only be able to understand the human instructions, but also to learn from the human guidance and improve its planning abilities gradually.



(a) Grasp a screw (b) Regrasp



(c) Place an aggregate (d) Put a part in



(e) Screw (f) Alignment

Figure 4: Examples of elementary operations.

The *planning* module on the scheduling level receives an EO from the *instruction understanding*. By referencing the *action memory*, the *planning* module chooses the corresponding basic primitive sequence for the operation. This sequence is a script of basic primitives for implementing the given EO. The task here includes planning of the necessary trajectories, choosing the right robot(s) and basic exception handling.

Sequences are executed by the *sequencer*, which activates different skills on the next *execution* level. The *planning* module also receives an event report that is generated by the *execution* level. If the event is a failure detection, the *monitoring* module is informed. In the normal operations, the *monitoring* module updates the action memory. It also detects the event failures. If it is found that the robot can re-do the operation, the *planning* module will try again. Otherwise, the *monitoring* module sends a request to the *dialog* module to ask the human communicator how to handle the exception and waits for an instruction. After the execution of each operation, the *knowledge base* is updated.

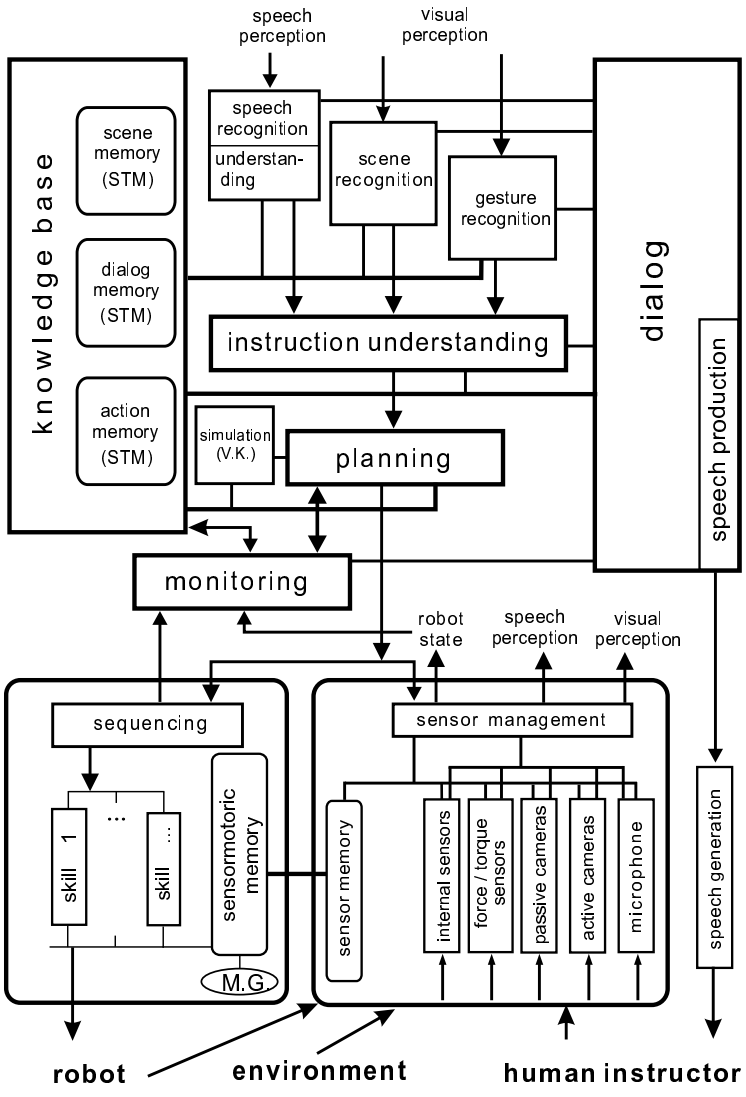


Figure 3: An architecture of the Situated Artificial Communicators for instruction understanding and execution.

5.3 Execution level

The *sequencing* module on the scheduling level uses the assembly skills provided by the execution level to perform a sequence. The complexity of the skills can range from opening the hand to collision-free control of the two arms to the meeting point. Advanced skills are composed of one or more basic skills. Generally, three different kinds of skills are classified: **(i)** Motor skills: *Open and close gripper; Drive joint to; Drive arm to; Rotate gripper; Move arm in approach direction; Move camera, etc.* **(ii)** Sensor skills: *Get joint; Get position in world; Get force in approach direction; Get torques; Check if a specific position is reachable; Take a camera picture; Detect object; Detect moving robot; Track an object, etc.* **(iii)** Sensorimotor skills: *Force-guarded motion; Vision-guided gross movement to a goal position; Visual servoing of the gripper to optimal grasping position, etc*.

5.4 Layered-learning

Learning the interplay of perception, positioning and manipulation as well as basic cognitive capabilities is the foundation of a smooth execution of a command sequence of a human instructor. If a command refers to an EO, the disambiguation of the instruction based on multimodal input is the key process. The autonomous sensor-based execution of these instructions requires adaptive, multi-sensor based skills with an understanding of a certain amount of linguistic labels. If complex instructions are used, however, the robot system should possess capabilities of skill fusion, sequence generation and planning. It is expected to generate the same result after a repeated instruction even if the situation has changed. The layered-learning approach is the scheme to meet this challenge.

Layered-learning is a hierarchical self-improving approach to realize multimodal robot control, in particular adaptive, multisensor based skills. Under this concept, tasks are decomposed from high to low level. Real situated sensor and actuator signals are located on the lowest level. Both self-supervised and reinforcement learning have been applied to the B-spline model [12] to realize most of the sensorimotor skills. Through task-oriented learning the linguistic terms to describe the perceived situations as well as robot motions are generated. Skills for manipulation and assembly are acquired by learning on this level using a neuro-fuzzy model. Furthermore, the learning results on the lower levels serve as the basis of the higher levels such as EOs, sequences, strategies, planning and further cognitive capabilities.

To learn the operation sequences automatically for two arms, we developed a method for learning cooperative tasks. If a single robot is unable to grasp an object in a certain orientation, it can only continue with the help of other robots. The grasping can be realized by a sequence of cooperative operations that re-orient the object. Several sequences are needed to handle the different situations in which an object is not graspable for the robot. It is shown that a distributed learning method based on a Markov decision process is able to learn the sequences for the involved robots, a master robot that needs to grasp and a helping robot that supports it with the re-orientation. A novel state-action graph is used to store the reinforcement values of the learning process.

5.5 Memories

To describe the knowledge base, both semantic and procedure knowledge are used. In our current implementation such knowledge is still hard-coded. It can be viewed as long-term-memory to a certain degree, which will be extended by learning approaches in our future research activities. Short-term-memories exist in perception modules, which are used for scene recognition, dialog preparation and action (sensorimotor functions). Learning of another important type of memories, the episodic memory, is preliminarily studied in the assembly scenarios.

According to the empirical investigations, the episodic memory represents one of the most important components of human intelligence. The reminding, mental simulation as well as planning use the episodic memory as the basis. The diverse multisensor data with large bandwidth of the robot such as vision system, joint angles, positions, force profiles etc., cannot be saved in their rough format for arbitrarily long time. Therefore, coding approaches based on appearances and features are suggested [9] for summarizing and generalizing experiences from the successfully performed operations. The multisensor trajectories and the motor signals are used for "grounding" the learned operation sequences.

5.6 Monitoring

Monitoring plays an important role to make an intelligent system robust. It is also used frequently by a human-being in manipulation and speaking, especially in a new environment or for a new task. Monitoring and eventually re-planning for repairing result in the non-linearity of the understanding-planning-execution cycle, but they represent one essential function in the cognitive architecture of a robot. Furthermore, it is meaningful to add a diagnosis function which can provide hypotheses about the reasons of diverse failures.

The unexpected events during the robot action can be for example: *A force exceeds a defined threshold; A camera detects no object; Singularity; Collision;* etc. If such an event happens, it is reported to the planning level.

6 Dialogue and Assembly Results

One example to build the "elevator control" aggregate of the aircraft out of three elementary objects by carrying out dialogues was studied. The objects were laid out on the table, and there were many more objects positioned in arbitrary order on the table than necessary. The HC had a complete image in his mind of what the assembly sequence should be. Alternatively, he could have used the assembly drawings in the construction kit's instructions and translated them into NL.

After the AC finding out if all objects are present and after going through an optional object naming procedure the HC input first triggers the action planner, which decides which object to grasp and which robot to use. Since the HC did not specify either of these parameters, both are selected according to the principle of economy. In this case, they are so chosen as to minimize robot motion. The motion planner then computes a trajectory, which is passed to the robots. Since there are enough bolts available, the AC issues its standard request for input once the bolt is picked up.

HC input results in the other robot picking up the slat. Before this may happen, however, it has to be cleared up, which slat to take. This involves the incorporation of the gesture recogniser. Then the screwing is triggered, involving the peg-in-hole module mentioned above followed by the screwing module. For reasons of space the subsequent steps of the dialogue have to omitted here; they show how error handling and many other operations can be performed – most of which humans are not aware of when they expect machines do do "what I mean". Fig. 5 shows two typical objects that can be built with the setup as developed up to now.

7 Future Work

Among many topics to be explored, some important ones can be listed as follows:

- The long-term-memory is learned from the short-term-memory so that symbols, sequences, names and attributes are anchored in the real sensor/actuator world.

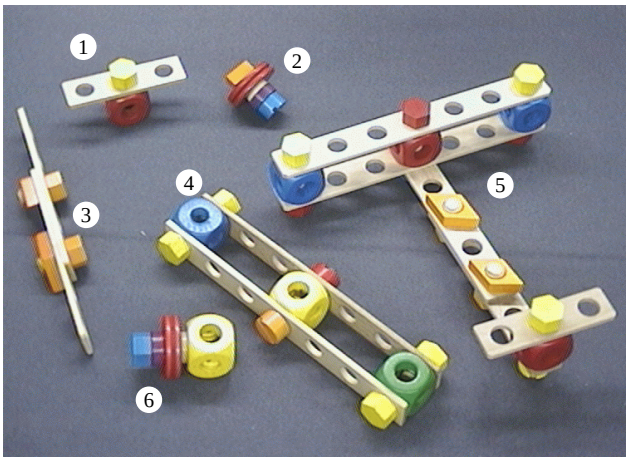


Figure 5: Sample aggregates made by our interactive assembly system.

- Methods need to be developed for increasing the capability and quality of reinforcement signals and fitness evaluation of the learning system. Active sensing and active manipulation can find their applications for these purposes.

- To enable the arbitrary transition between digital measurements and concepts, symbolic sparse coding, granular computing, fuzzy sets and rough sets will be investigated and integrated.

- Action sequences learned on the basis of verbal and visual instructions and summarization need to be built into an appropriate representation so that they can be generalized for analog or even new tasks.

- Learning on the higher level should be conducted to select action strategies and to generate intelligent dialogs. This will need the tight integration of more components and more knowledge shown in Fig. 3.

- More functions such as a motivation or creation module need to be added in the architecture so that the robot's initiatives can be used instead of passive acceptance of instructions.

Acknowledgment

This research is supported under grant SFB 360 by DFG, the German Research Council.

References

[1] J. S. Albus. The engineering of mind. In *Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior: From Animals to Animats*, September 1996.

[2] R. Bischoff and V. Graefe. Integrating vision, touch and natural language in the control of a situation-oriented behavior-based humanoid robot. In *IEEE International Conference on Systems, Man, Cybernetics, Tokyo*, 1999.

[3] R. A. Brooks, C. Breazeal, M. Marjanovic, and B. Scassellati. The Cog project: Building a humanoid robot. In C. L. Nehaniv, editor, *Computation for Metaphores, Analogy and Agents*, volume 1562 of *Lecture Notes in Computer Science*, pages 52–87. Springer, 1999.

[4] A. Clark and R. Grush. Towards a cognitive robotics. *Adaptive Behavior*, 7(1):5 – 16, 1999.

[5] G. Lakoff. *Women, Fire, and Dangerous Things: What Categories Reveal About the Mind*. University of Chicago Press, 1990.

[6] R. Moratz, H. Eikmeyer, B. Hildebrandt, A. Knoll, F. Kummert, G. Rickheit, and G. Sagerer. Selective visual perception driven by cues from speech processing. In *Proc. EPIA 95, Workshop on Appl. of AI to Rob. and Vision Syst.*, TransTech Publications, 1995.

[7] R. T. Pack, M. Wilkes, G. Biswas, and K. Kawamura. Intelligent machine architecture for object-based system integration. In *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, June 1997.

[8] K. R. Thorissen. *Communicative Humanoids - A Computational Model of Psychosocial Dialogue Skills*. PhD thesis, MIT Media Lab., 1997.

[9] Y. von Collani, J. Zhang, and A. Knoll. A general learning approach to multisensor based control using statistical indices. In *Proceedings of the 2000 IEEE Conf. on Robotics and Automation, San Francisco, California, April 2000*.

[10] J. Weng, C. H. Evans, W. S. Hwang, and Y.-B. Lee. The developemental approach to artificial intelligence: Concepts, developmental algorithms and experimental results. In *In Proc. NSF Design & Manufacturing Grantees Conference*, 1999.

[11] T. Yamada, J. Tatsuno, and H. Kobayashi. A practical way to apply the natural human like communication to human-robot interface. In *Proceedings of 10th IEEE International Workshop on Robot and Human Communication*, pages 158–163, Bordeaux-Paris, September 2001.

[12] J. Zhang and A. Knoll. *A Neuro-Fuzzy Learning Approach to Visually Guided 3D Positioning and Pose Control of Robot Arms*. In "Biologically Inspired Robot Behavior Engineering", edited by R. Duro, J. Santos and M. Grana, Springer Verlag, 2001.

[13] J. Zhang, Y. von Collani, and A. Knoll. Interactive assembly by a two-arm robot agent. *Journal of Robotics and Autonomous Systems*, 29:91–100, 1999.