

Intuitive Teaching and Surveillance for Production Assistants

S. Estable, I. Ahrns, H.G. Backhaus, O. El Zubi, R. Münstermann

Astrium Space Infrastructure

P.O. Box 28 61 56

28361 Bremen, Germany

E-mail: stephane.estable@astrium-space.com

Abstract

The need for production assistants will increase in the next years. They will allow to fulfill new factory requirements like the production of small series, the reduction of the innovation cycles and the optimization of the factory workload. The possible components of such a production assistant, dedicated to object manipulation tasks, has been investigated by Astrium in the project MORPHA¹. Two features seem to describe such an assistant system: intuitive teaching and surveillance. According to this statement, three main components have been specified and implemented: pose estimation skills, intuitive trajectory generation and surveillance for workspace sharing. These components will be described and the results evaluated.

1 Introduction

A study from the ISI Institute [1] reports that the degree of automation in German companies is decreasing. According to the study, the interviewed companies wish more flexibility in automation solutions. The adaptation of high automated production units to different production volumes and new products is very expensive. The factors when developing a new product are still the same: product meets needs, high quality and low price. But a new factor is arising: customization. The customer wants personalized products [2], not only bulk goods.

The ISI study lists three requirements for more efficient production units turned to productivity and customization:

- Produce small series: for a new product, investments in production tools have to be lower.
- Reduce innovation cycles: for a new product, the adaptation time of production tools has to be shorter.
- Optimize factory workload: a production tool is able to deal with different kind of products.

Which kind of technical system with which functionalities can reach these requirements?

Production assistants have been investigated in the project MORPHA like the manufacturing assistant from DaimlerChrysler [3]. In order to built assistant systems very important efforts have to be made in the man-machine interaction. Robots have to be taught by human, on the other hand human beings and robots have to co-exist in a work cell. These two features, intuitive teaching and surveillance, seem to be important parts of a production assistant dedicated to object manipulation for assembly, quality check or palletizing. At Astrium three main components have been specified for the realisation of a manipulation assistant:

- Quickly adapt skills to new objects: Teachable 2D/3D object recognition and pose estimation.
- Quickly and intuitively generate or adapt robot programs: Programming by demonstrating.
- Ensure safety while human being and robot share the same workspace: Surveillance based on range images from the Astrium's laser camera [4].

In this paper we intend to show possible ways of realization of the three sub-systems. They are described in the sections 2, 3 and 4 respectively. In section 5 the results are discussed and summarized.

2 Pose estimation skills

An object manipulation system has to deal with different types of object classes: the objects to be handled but also the containers. The first class represents a large variety of shapes, sizes or colours, the latter is more restrictive due to simpler geometric constraints. In both cases the pose estimation methods have to meet the following criteria:

- easy and fast teaching,
- handle simple to complex objects,
- support arbitrary backgrounds and
- robust against occlusion.

Two main approaches have been implemented according to the object complexity: segmentation-free and segmentation-based pose estimation. The extraction of object specific features and methods based on a segmentation/reconstruction scheme have been avoided for complex objects. Complex objects are not suitable for

¹ This research was partly sponsored by the German Ministry for Education and Research under the project MORPHA, Intelligent Anthropomorphic Assistance Systems.

segmentation and reconstruction-based teaching leads to higher teaching times.

2.1 Segmentation-free pose estimation

2.1.1 General design

The implemented segmentation-free methods are based on standard approaches like template matching. They include the following steps:

- Compute distance transform images on edges [5],
- Generate hypotheses on raw object location: edge matching with full search in the distance transform images on different hierarchy levels,
- Check hypotheses based on region matching to get raw pose estimation,
- Accurate pose estimation for the best hypotheses based on line fitting or correlation,
- Compute the grasp position.

Two kinds of teach-in strategies have been selected:

- Feature based teaching: the shape of the object is selected in the image by the operator,
- View based teaching: the object is automatically presented to the sensor in relevant poses.

The distance transform method [5] is a very efficient method for computing the Euclidean distance from one point in the image to the nearest edge. The sum along all points of the model gives for a position of the model in the image the Euclidean distance of the model to the image.

2.1.2 2D pose estimation

The object model is generated with the feature-based teaching strategy: the relevant contours of the object are selected in the image for finding the raw pose estimation (one gets a segment list), templates are selected for hypotheses check, main lines or areas are marked for computing the accurate pose estimation and the grasp position is set.

The recognition and pose estimation is achieved as described above on grey images: the list of segments, which is part of the model, is matched at every position in the distance transform image (edge matching by computing the distance between the current position of the model and the edges), the generated hypotheses are checked with the templates stored in the model and the accurate position is computed for the best hypothesis based on line fitting or correlation.

The first results on different types of caps show a very high recognition rate (99%) and a high pose accuracy (assembly position 1/10 mm, noise 1/20 pixel, rotation 1/50 degree) on non uniform backgrounds. The different objects have been taught with the same tool (Figure 1).

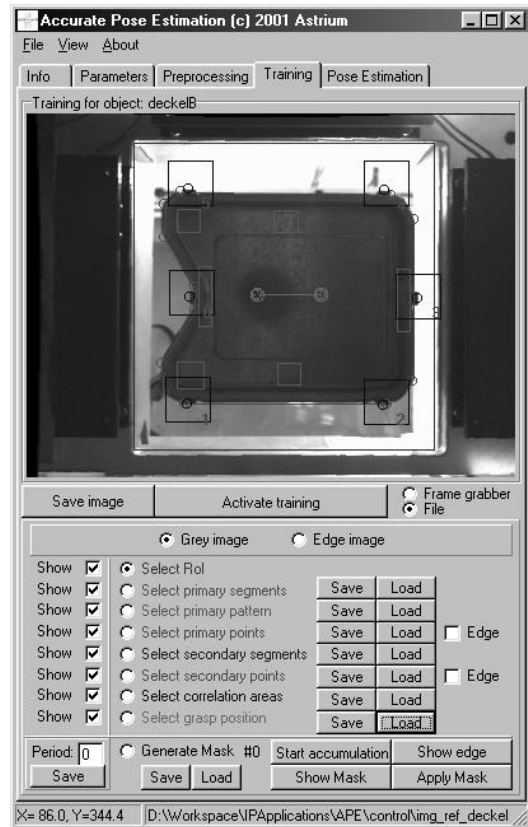


Figure 1. Teach-in tool for the 2D pose estimation.

2.1.3 3D pose estimation

The object model is generated with the view-based teaching strategy: for each application relevant position of the object a set of templates encoding both the edges and the region corresponding to the object in the range image are generated.

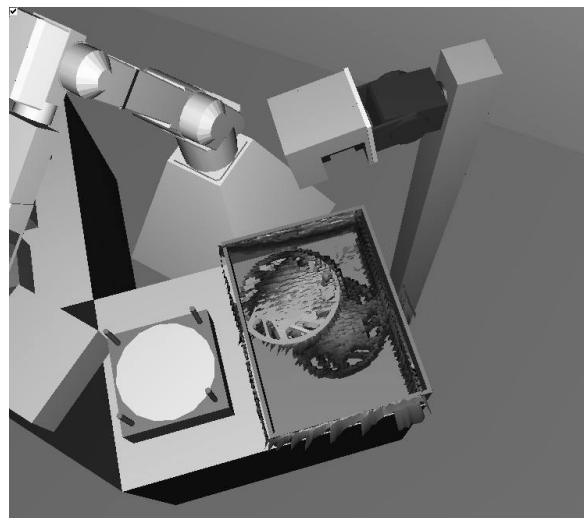


Figure 2. Real data of the 3D-Laserscanner with pose estimation result in a virtual environment.

The dense range images are acquired by a 3D-Laserscanner. The edge templates are matched in the distance transform images, which are computed from the range image. The best matches generate hypotheses on the raw object location. Range information of the image is used to pre-select a subset of templates that fit this valid range. Hypotheses are validated applying a region match. Actually no accurate pose estimation is implemented for this 3D pose estimation. A possible method could be Iterative Closest Point [6].

Pose estimation results are depicted on figure 2. The recognition rate reaches 95% and the pose accuracy is around 20 mm.

2.2 Segmentation-based pose estimation

In the field of automation an important number of applications deals with containers. This class of objects contains geometrical features like planes, lines and edges. For this case segmentation-based methods are suitable and an effective way to solve the problem of object pose estimation.

The containers are sensed by a 3D-Laserscanner. The segmentation-based pose estimation consists of the following three modules:

- edge extraction,
- line extraction and
- pose determination.

The edge extraction searches for discontinuities in the 3D Cartesian data and the line extraction builds lines from the extracted edges (Figure 3). The pose determination matches the extracted lines with the lines stored in the container model (Figure 4).

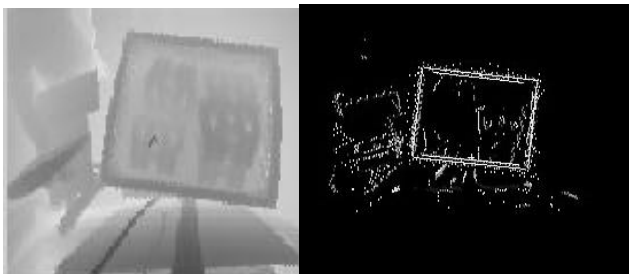


Figure 3. Range image of a container (left) and corresponding extracted edges and lines (right).

The quality of the lines depends on the generated edges, which in turn depends on the resolution of the 3D scan. The pose of the containers is estimated with 20 mm accuracy in the position and 1 degree in the orientation.

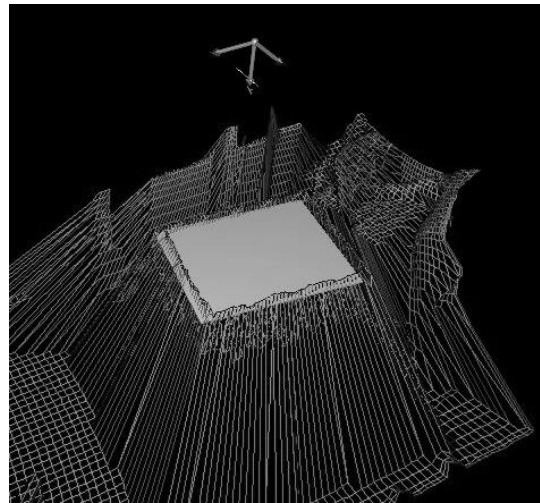


Figure 4. Pose estimation in the Cartesian data.

3 CyberTeachTool

3.1 Programming by doing with the CyberTeachTool

The principle of the CyberTeachTool (CTT) is to generate trajectories and robot programs in an intuitive manner of interaction using a teach tool, speech recognition and three dimensional stereoscopic visualisation instead of computer keyboard and mouse. Leaving the two dimensional plane of an computer monitor leads directly into the real world of programming robot applications.

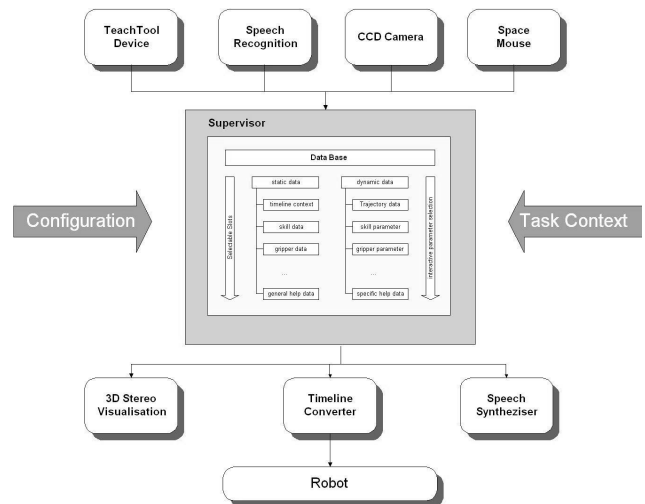
CyberTeachTool as a frame element is the connecting link between robot trajectory generation, skill integration, visual timeline validation and finally timeline execution on the robot target system.

But what is the difference to similar systems? Ehrenmann et al. in [7] and Kheddar et al. in [8] present a programming by demonstration system that emphasizes the interpretation of what have been done by the human demonstrator. This is the main difference with the CyberTeachTool from Astrium where the different inputs are not interpreted but directly stored in a data structure. In [9] Freund et al. present a space robotics application based on a resource-based action planing for an intuitive control and supervision of a robot arm. The automatic program generation and monitoring allow to handle real and simulated instances of the system at the same time for several robots.

3.2 Components and Architecture

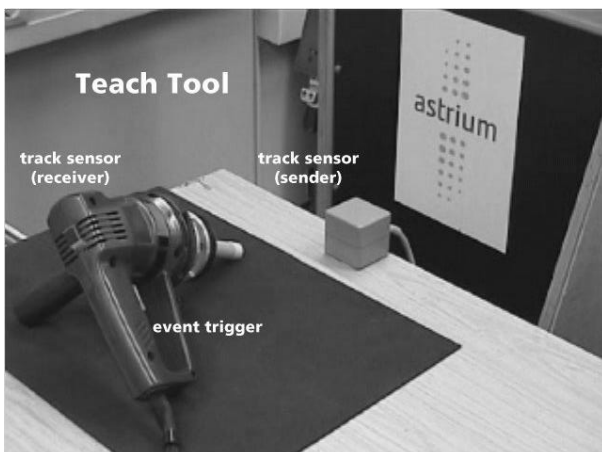
Basically CyberTeachTool consists of the Head Set:

- Stereo glasses for 3D stereo visualisation
- Head tracker
- Speech recognition / synthesis



and the Teach Tool itself:

- Track sensor (sender / receiver)
- Event trigger
- Interface to robot



3.3 Generation of timelines

The robot application programmer will be supported in each step of his programming task. At first he will generate trajectories in the workspace of the robot. This is simply done with the Teach Tool device (Figure 6) by moving it to the desired pose, saving the current pose and repeating these steps until an appropriate trajectory is available.

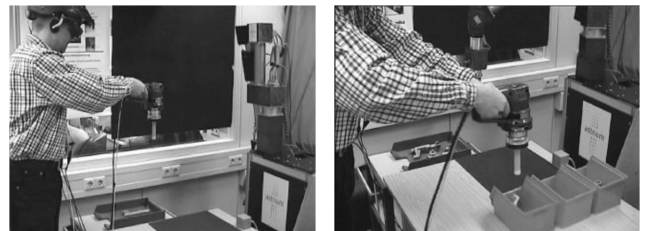


Figure 6. Trajectory generation

All external devices are connected to the computer (Figure 5). The supervisor module as the core of the system is responsible for managing the whole teaching process. The configuration of the system, i.e. availability of skills, objects to be handled by dedicated skills as well as skill help information will be loaded during start-up.

During the teaching process context sensitive information is available. The user is able to make his decisions on base what he sees and hears. The 3D-stereo visualisation allows for a realistic "look and feel" of trajectories and selections made.

At least the generated timeline will be executed stepwise on the dedicated robot. The timeline converter compiles the timeline information into a script format which can be executed on the robot target system.

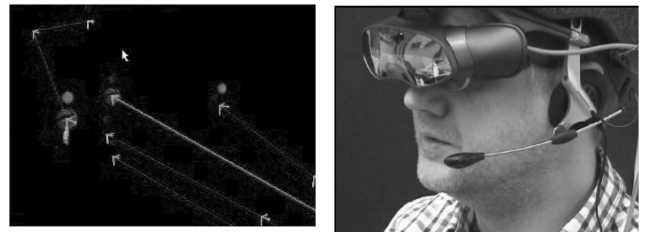


Figure 7. Virtual timeline programming by voice commands

The user acts as a decision maker. Connecting trajectories, integrating skills, specifying objects to be selected or gripper commands to grasp selected objects, all is done by commanding the system by voice (Figure 7).

A big advantage of the CTT is that visual programming allows for instant checking of programming results (Figure 8).

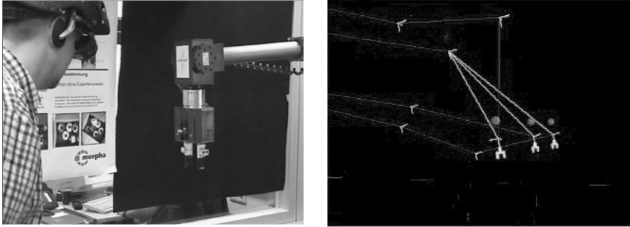


Figure 8. Checking the timeline in the real scene

The next step leads to timeline execution, either step by step to assure correct robot actions or in one complete run (Figure 9).



Figure 9. Stepwise testing of timelines

The final step yields generating a robot program which is executable on the robot target system. As this compilation is robot hardware dependant, the dedicated script translation library has to be linked to CTT.

3.4 Results and evaluation

The CyberTeachTool has been developed as an integral part of MORPHA. The aim was to show intuitive robot programming capabilities. Especially the usage of the Teach Tool, the three dimensional stereoscopic visualisation and the speech recognition supports the robot application programmer in the generation and adaptation of robot programs. The calibration of the vision system [10] allows for realistic scaling of virtual objects to the real scene. Changing the tracking device will increase the spatial accuracy of the system as well as compensating the electromagnetic sensitivity.

At least, the major goal "Let 2D go Space!" could be achieved.

4 Surveillance based on range images

Robot working cells are strictly separated from working areas where humans operate. Till this day, if a worker enters the robot's working cell, the robot must stop immediately, in order to protect the worker from being hurted by the robot. Unfortunately, this strategy of stopping the robot leads to robot working cells which do not allow any interaction between a robot and a human worker. In this section we present first approaches and experiments to overcome this problem while the safety of the worker is still preserved.

Video surveillance is a problem which is discussed by many authors in the computer vision community. In most of the cases they use standard CCD cameras and try to detect the pose of human bodies (e.g. [11]). Gavril presented in [12] a survey on this topic. In contrast to these approaches, we keep the robot working cell under surveillance by using a solid-state laser range camera [4, 13]. From this camera we obtain dense range images at a frame rate of 7 Hz. By superposing the range data from the camera and the range information from a virtual model of the robot working cell, we are able to simply separate the data coming from the robot and the data coming from an intruder, e.g. the human worker. If the 3D points of the worker's range information enter a danger zone around the robot, the speed of the movement of the robot is reduced, or the robot is even stopped.



Figure 10. Experimental robot working cell.

4.1 Sensing the robot working cell

In our experimental setup our laser range camera is mounted above the working cell such that the robot's manipulation area and the workspace of the human are both in the camera's field of view. This situation is depicted in figure 10, where one can see the sensor in the upper left corner of the image. Figure 11 shows the range image of the robot's gripper sensed by the laser camera. The main problem now arises from the fact that not only an intruder, but also the robot itself produces range data

which are naturally lying in any danger zone which can be defined around the robot. For that reason, we need to distinguish between range data coming from the robot itself and the remainder.

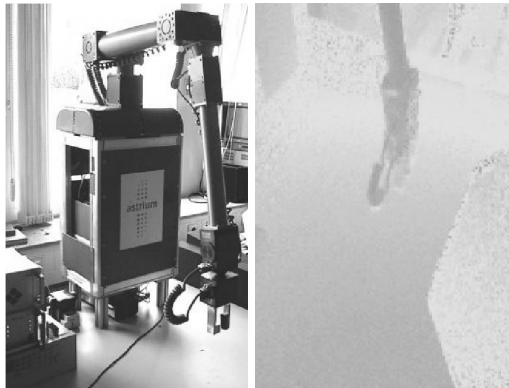


Figure 11. Robot and range image.

4.2 Robot segmentation

In order to segment those regions in the range image that belong to the robot, we use a dynamic 3D model of the robot which can be rendered by standard render engines like OpenGL. The 3D model is regularly updated by the real joint angles of the robot. By determining the transform between the robot base and the sensor system by standard external camera calibration techniques, we are able to build a virtual scene of the robot model which is observed by a virtual camera. The virtual camera is placed in the virtual world such that the virtual image corresponds to the real camera image. The virtual range image is obtained by reading OpenGL's Z-buffer.



Figure 12. Virtual (left) and real (right) range image.

Figure 12 depicts a virtual range image of the robot base with a part of the end effector and shows the corresponding real range image. By comparing the virtual and the real range images, we are now able to segment the robot in the range image. The same trick is also applicable to other parts in the scene which can be modelled in the VR world. For instance we also model a manipulation area on top of the table in front of the robot where the robot is

allowed to manipulate several objects. In order to filter out the range data coming from the possibly unknown objects in front of the robot, we model the manipulation area as a small box on top of the table. Range information coming out of this box is filtered out. In order to prevent the human worker to enter the manipulation area, we define a further box lying on top of the manipulation area. If 3D points which were sensed by the laser camera and then filtered fall into this box, the system detects someone entering the manipulation area and the robot must stop.

4.3 Obstacle Detection

We now have filtered versions of the range image where the following objects can be distinguished: the robot (dynamically updated), the manipulation area, and data coming from any intruder which might be an obstacle for the robot. Static background information can be trivially filtered out by simply subtracting a background image.



Figure 13. Range image and corresponding segmented range image with obstacle.

The 3D points which do not belong to the background, neither to the robot, nor to the manipulation area are points which are potential obstacles for the robot. Figure 13 shows an example of obstacle points after filtering out the robot and the manipulation area. The arm of the worker entering the robot's workspace is well detected.

4.4 Danger Zones

Using the 3D points of the potential obstacles, a possible collision between the intruder and the robot can be detected. For that purpose, we define danger zones around the robot model. These danger zones can be built of any arbitrary geometric primitive, like cubes, cones, and spheres. Spheres are the most simple primitives which allow the most efficient collision check between the primitive and a 3D point. Due to the rotational symmetry of the sphere, all transforms are simple translations. Figure 14 shows the virtual robot where spherical danger zones are defined around every joint of the robot.

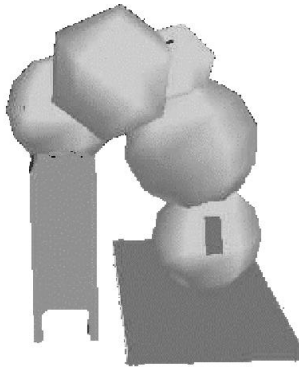


Figure 14. The virtual robot and spherical danger zones.

4.5 Experiments

The following section finally presents some experimental results. Figure 15 depicts an arm entering the workspace of the robot. The 3D points coming from the arm are filtered out and the collision check between these points and the spherical danger zones raises an alarm (visualized by different colors of the spheres).

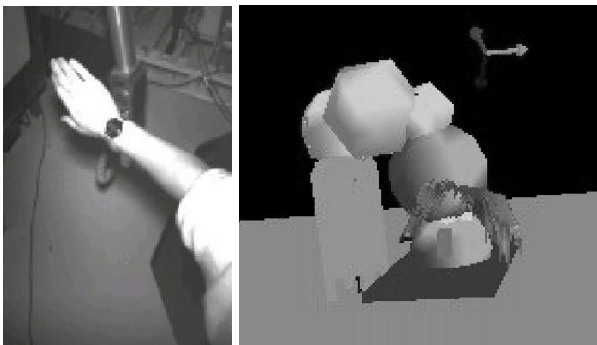


Figure 15. Detection of an obstacle entering the danger zone of the robot.

Figure 16 depicts a slightly different situation, where the worker enters the manipulation area. The collision with the manipulation area is also detected and the robot is forced to stop the movement. After the worker leaves the manipulation area, the robot continues the work.

The situation when nothing disturbs the robot work is shown in figure 17. The robot moves and the range data coming from the robot itself is left out for the collision check.

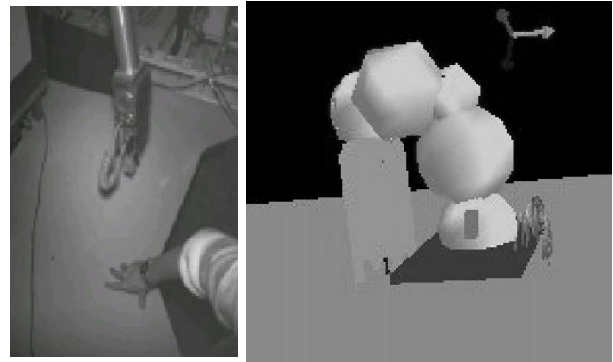


Figure 16. Detection of an obstacle entering the manipulation area.

It is even possible that the robot enters its own manipulation area where objects are manipulated. In this case no alarm is raised from the surveillance system, since no obstacles are detected. Collisions between the robot and the manipulation area are explicitly allowed.

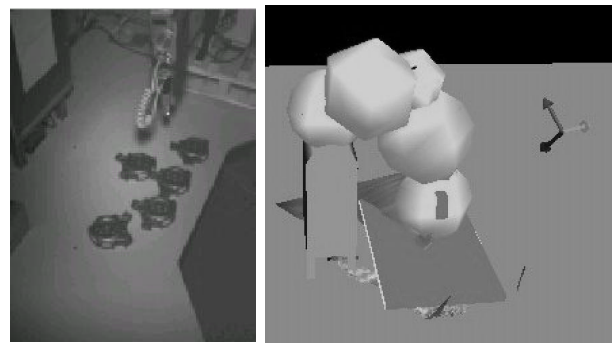


Figure 17. The robot enters the manipulation zone. There is no obstacle and the robot is allowed to move.

5 Conclusion

The discussion of requirements focussing on the definition of a production assistant yields for: quick skill teaching, intuitively generation of robot trajectories and granting the safety for human beings co-operating with robots. In this paper we are proposing three sub-systems which should comply these requirements. The evaluation of the first results shows real chances and also important fields of improvement.

The 2D and 3D pose estimation skills have demonstrated very good performances in recognition rate and accuracy. The teach-in tools have proved their ability to flexibly generate object models. A possible use in real industrial systems for assembly or palletizing is currently considered.

The CyberTeachTool system from Astrium for programming by doing has demonstrated the real use of this function. Intuitively marking trajectories in the workspace of the robot, including skills and stepwise

testing the robot program only with a teach tool, stereo glasses and speech recognition make the programming task more efficient. The tracking system performs in this application the main challenge. We experienced that a much better accuracy, a larger workspace and an environment independent tracking sensor technology is required for introducing the system into field applications. [14] presents robust solutions of the tracking problem.

The surveillance based on dense range images has shown a good performance. The obstacle detection algorithm allows a robust and fast detection in the vicinity of the robot. The main challenge is the 3D sensor, which has to deliver for such a safe critical application dense range images at a much higher frame rate.

Combining all above mentioned elements into an integrated manipulation assistant is the next logical step. We see the introduction of manufacturing assistants in factories as a sensible step to more production flexibility.

References

- [1] G. Lay, E. Schirrmeister, "Sackgasse Hochautomatisierung? Praxis des Abbaus von Over-engineering in der Produktion", in *Mitteilungen aus der Produktionsinnovationserhebung*, Nummer 22, Fraunhofer Institut Systemtechnik und Innovationsforschung, Mai 2001.
- [2] F. Piller, "Kundenindividuelle Massenproduktion: Die Wettbewerbsstrategie der Zukunft", in Carl Hanser (Eds.), 1998, 410 pages, ISBN: 3-446-19336-7.
- [3] Stopp, A. Horstmann, S. Kristensen, S. and Lohnert, F. "Towards Interactive Learning for Manufacturing Assistants", In *Proc. of the 10th IEEE Inter. Workshop on Robot-Human Interactive Communication (ROMAN'01)*, Paris, France, September 2001, pp. 18.-21.
- [4] W. Schröder, E. Forgber, G. Röh, "Laser range camera applications." In *Proc. of the fifth ESA Workshop on Advanced Space Technologies for Robot Applications*, Noordwijk, The Netherlands, 1998.
- [5] D. Gavrila, "Multi-feature Hierarchical Template Matching Using Distance Transforms", in *Proc. of the IEEE International Conference on Pattern Recognition*, Brisbane, Australia, 1998, pp. 439-444.
- [6] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes", in the *IEEE Trans. Pat. Anal. and Mach. Intel.* 14(2), Feb 1992, pp. 239-256.
- [7] M. Ehrenmann, O. Rogalla, R. Zöllner, and R. Dillmann, "Teaching Service Robots Complex Tasks: Programming by Demonstration for Workshop and Household Environments", in *Proc. of the IEEE Int. Conf. on Field and Service Robotics 2001 (FRS)*, Finland 2001.
- [8] A. Kheddar, C. Tzafestas, P. Coiffet, T. Kotoku and K. Tanie, "Multi-robot teleoperation using direct human hand actions", in *Advanced Robotics*, Vol. 11, No. 8, pp. 799-825, 1998.
- [9] E. Freund, K. Hoffmann, and J. Rossmann, "Application of automatic action planning for several work cells to the German ETS-VII space robotics experiments", in *Proc. of the 2000 IEEE Int. Conf. on Robotics & Automation*, San Francisco, CA, pp.1239-1244, April 2000.
- [10] R. Grasset, X. Decoret, and J.D. Gascuel, "Augmented Reality Collaborative Environment: Calibration and Interactive Scene Editing", in *VRIC, Virtual Reality International Conference*, Laval Virtual, 16-18 May, 2001.
- [11] D. Gavrila and L. Davis, "3D-model-based tracking of human in action: a multiview approach.", in *Proc. of IEEE Conference of Computer Vision and Pattern Recognition (CVPR)*, San Francisco, 1996, pp. 73-80.
- [12] D. Gavrila, "The visual analysis of human movement. A survey.", In Roberto Cipolla/Alex Pentland (Eds.) *Computer Vision for Human-Machine Interaction*, pp. 1-6;12-25.
- [13] S. Vignier, "A real-time system for dynamic obstacle detection in the environment of manipulating robots based on range images.", Master Thesis, ESIEE Paris, 2001.
- [14] S. Lehmann, "A hybrid tracking approach for augmented reality applications", in *Proc. of International Scientific Conference on Work with Display Units*, Berchtesgaden, Germany, 2002, pp.384-386.