2012 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 7-12, 2012. Vilamoura, Algarve, Portugal

Spatial Programming for Industrial Robots based on Gestures and Augmented Reality

Jens Lambrecht[1] and Jörg Krüger[1,2]

Abstract— **The presented spatial programming system provides an assistance system for online programming of industrial robots. A handheld device and a motion tracking system establish the basis for a modular 3D programming approach corresponding to different phases of robot programming: definition, evaluation and adaption. Static and dynamic gestures enable the program definition of poses, trajectories and tasks. The spatial evaluation is done using an Augmented Reality application on a handheld device. Therefore, the programmer is able to move freely within the robot cell and define the program spatially through gestures. The camera image of the handheld is simultaneously enhanced by virtual objects representing the robot program. Based on 3D motion tracking of human movements and a mobile Augmented Reality application, we introduce a novel kind of interaction for the adaption of robot programs. The programmer is enabled to interact with virtual program components through bare-hand gestures. Such sample forms of interaction include translation and rotation applicable to poses, trajectories or tasks representations. Finally, the program is adapted according to the gestural changes and can be transferred from the handheld device directly to the robot controler.**

I. INTRODUCTION

The demographic change and a general shortage of skilled workers are key challenges for manufacturing in western industrialized countries. In contrast to an ageing society is a new generation of "digital natives", young people who are familiar with using new technologies (e.g. computers and handhelds) intuitively and highly efficiently. One can meet the challenges of an ageing workforce both by taking preventive measures and through intelligent assistance systems, which support the worker in manual production processes toward ergonomics and efficiency. Hence, the application of assistance systems could enable a sustainable design of manual tasks [1].

Regarding industrial robotics, manual online programming requires a high degree of expertise. Due to the time-consuming and complex programming process, small and medium-sized enterprises have reservations about investing in an industrial robot [2]. Thus, multimodal communication has increasingly become the subject of scientific approaches for novel programming techniques in industrial robotics. Multimodal human robot interaction includes natural human-to-human communication, which mostly consists of speech (auditory) and gestures (visual). In terms of ergonomics, the main objectives of using natural communication are a shortening of both training times and operating cycles. Multimodality, i.e. a use of multiple communication channels simultaneously or successively, plays an important role in the design of intuitive control systems. Numerous scientific publications, e.g. [3], [4], [5], [6], showed that multimodal control systems are more efficient and user friendly in comparison to conventional systems. However, an essential requirement is an adequate system design, which is related to the application and the user. Multimodal control systems typically use gestures, e.g. hand and finger gestures, haptics and speech for communication. Relevant publications in the field of gestural control of industrial robots can be found in [7], [8]. Another type of interaction between humans and machines is visualization. Unlike conventional forms of visualization, Augmented Reality (AR) enhances a camera image by adding spatially related information. In the field of industrial robotics, the user can be supported e.g. by providing spatial information about the robot programs or coordinate systems. This means that, poses, trajectories and information on task level can be visualized within the real robot environment. Similar to simulation systems, a virtual robot can run the program with regard to accessibility and collision control. Thus, in AR simulation-based features are carried out within the real environment [9].

In this paper we introduce a spatial programming system for industrial robots, including different modules for gesture-based definition of poses, trajectories and tasks. Additionally, the system covers program evaluation in an AR applications as well as program adaption through spatial interaction with virtual objects representing the robot program. The paper is organized as follows: section 2 outlines previous research of the authors and the current state of the art. Section 3 describes the overall framework of spatial robot programming, taking into account different levels of robot programming. Section 4 introduces general methods for the realization of the programming approach while section 5 describes a sample implementation. Finally, section 6 presents our conclusions.

II. RELATED WORK

Our current project is based on previous research on the gestural control and programming of industrial robots [10] and an unified control layer [11], which enables robot control via arbitrary devices, e.g. through smartphones. More realted work in the field of robotics and automated assembly was on physical human robot interaction [12], on work step recognition [13] with 3D cameras and on assitance

[1]Both authors are with the Department of Industrial Automation Technology, Institute for Machine Tools and Factory Management, Berlin Institute of Technology, Pascalstr. 8-9, Berlin, Germany lambrecht at iwf.tu-berlin.de
[2]Author is with Fraunhofer Institute for Production Systems and Design Techononlgy (IPK), Berlin, Germany

systems [1]. In the following, we give a short outline on the state of the art of multimodal industrial robot control and programming with an emphasis on gestures and AR.

In 2003 the research project Morpha [5] was completed. The project was funded by the German Federal Ministry of Education and Research and consisted of a broad consortium of robot manufacturers, research facilities and application companies. The main research focused on the development of interactive and innovative technologies for human-robot-interaction through natural communication. The research results are still the basis for many research projects in the domain of multimodal control for service and industrial robots. The consortium introduced a rudimentary gesture-based motion control, task-oriented programming as well as a static AR application for the visualization of coordinate systems and axial moving directions.

Meanwhile, many international research projects dealt with the application of AR to industrial environments. For a broad overview of industrial AR applications, technical requirements and implementations, refer to [14]. In the following we give an overview on recent scientific publications strongly related to our project.

W. Vogl [15] addressed gesture-based definition of trajectories as well as AR-based evaluation of welding tasks. Moreover, a spatial interface was created to interact with virtual displays, which are projected onto material surfaces by means of a laser projector. This has been used to adjust parameters for the welding job. From our point of view, disadvantages of this system are: the definition of trajectories by an additional tools ("Magic Pen") and the use of an expensive motion-tracking system tracking fiducial markers. Furthermore, spatial program adaption is limited to these 2D displays.

Akan et al. [6] introduced an AR application with the objective of task-oriented programming of industrial robots. The camera is fixed in the workspace of the robot or mounted to the robot. Moving virtual objects in a graphical user interface enables the definition of assembly tasks.

The Augmented Reality and Assistive Technology Laboratory at the National University of Singapore studied different AR applications. They have recently developed a bare-hand gesture-based interaction with virtual 3D objects for AR applications [16]. The gesture recognition is based on 2D data from a single camera. A possible extension of 2D to 3D gesture interaction through a stereo camera system is mentioned, but not implemented yet. Another publication [17] considers a definition of assembly tasks and trajectories for robot programming based on the movement of a marker-cube in an AR application. However, regarding industrial robotics neither interaction based on 3D sensor data, nor approaches for gestural adaption of different program components were presented.

III. SPATIAL ROBOT PROGRAMMING

The overall objective of our project is to establish suitable modalities for intuitive industrial robot programming. The presented spatial programming system provides an efficient assistance system for online programming of industrial robots. In contrast to current state of the art technologies, our spatial system comprises the support for different phases of the programming process as well as different levels of robot programming. To this end, we put an emphasis on gesture-based, markerless definition of poses and trajectories as well as the gestural manipulation.

Furthermore, we aim for a mobile AR solution on conventional handheld devices. Thus, the programmer is not confined to a fixed computer workstation but can define poses, trajectories and tasks, on-the-move, as well as watching the spatial arrangement of the program components and manipulate single objects. The three sub-modules of the system correspond to different phases of robot programming: definition, evaluation and adaption (see Fig. 1). Single programming modules are also applicable in combination with conventional online and offline programming methods.

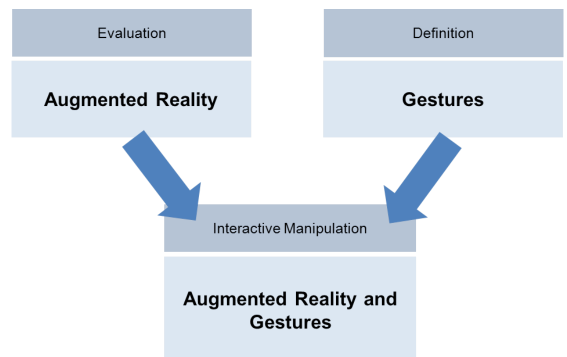

Fig. 1. Modalities for spatial online-programming.

A. Levels of Robot Programming

In terms of a holistic approach toward robot programming, we consider different programming levels for spatial interaction. Thus, the forms of interaction presented in this paper are applicable to different levels of programming: low-level and high-level programming. For the purpose of task level programming, virtual objects cover task representation, e.g. workpieces for Pick'n'Place tasks. These virtual workpieces can be manipulated, e.g. by gestural snapping and moving in space. Besides Pick'n'Place tasks, an extension of graphical task level representations and spatial interaction principles is possible due to a modular structure. In a sample case of line welding, the programmer may define the path at a workpiece by drawing a virtual line with his finger. Subsequently the welding line and poses are visualized in the AR application on the handheld, where the user can manipulate the virtual objects, e.g. to manipulate tool orientation of single poses. Regarding this specific case and the required accuracies the application of additional sensors would be adequate.

In summary, this means that the programmer is enabled to switch between different levels of robot programming, due to his personal or task specific requirements. In comparison with conventional prorgramming systems, this is what makes our spatial system more flexible and efficient for the definition, evaluation and adaptation of robot programs. Due to the insufficient accuracies of most markerless motion tracking systems, the application of additional sensors is essential for specific tasks.

B. Gestural Program Definition

Regarding spatial program definition, we introduced an approach for industrial robot programming in [10] using a markerless motion tracking system. The appproach already covers an intuitive gesture-based system for the definition of poses and trajectories by gestures, e.g. pointing gestures for poses. Furthermore the programmer can define complex trajectories by natural movements. Fig. 2 outlines the principle of defining poses, trajectories and tasks through pointing gestures and finger movements.

Fig. 2. Spatial program definition through gestures on different levels of industrial robot programming including poses (left), trajectories (center) and tasks (right).

C. Program Evaluation

The evaluation of the robot program covers visualization and simulation and is possible through an AR application on a handheld device. Therefore, the programmer is capable to move freely within the robot cell, while the camera image is enhanced by spatial representations of the interpreted robot program. In Fig. 3 one can see the visualization principle on the different levels of program representation. Additionally, a virtual robot can run the program.

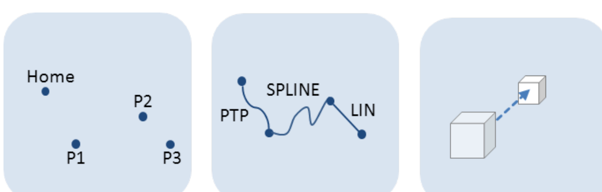

Fig. 3. Levels of spatial visualization in the AR application for robot programming including poses (left), trajectories (center) and tasks (right).

D. Spatial Program Adaption

Based on the motion tracking of human movements and a mobile AR environment there arises a novel kind of interaction for the manipulation of the robot program. Using the AR visualization the programmer interacts through bare-hand gestures in front of the handheld device. Fig. 4 illustrates some fundamental command gestures for elementary object interaction. Snap and release gestures intend to start and end an object specific interaction. Poses, trajectories and objects can be manipulated through 3D movements of the hand or fingers, i.e. the programmer can translate or rotate the virtual

Fig. 4. Sample set of simple gestures for bare hand interaction with virtual objects. Snap (left), release (center) and pointing (right).

object. Subsequently the robot program is adapted automatically according to the change through spatial interaction. Fig. 5 illustrates some conceivable methods of spatial object manipulation.

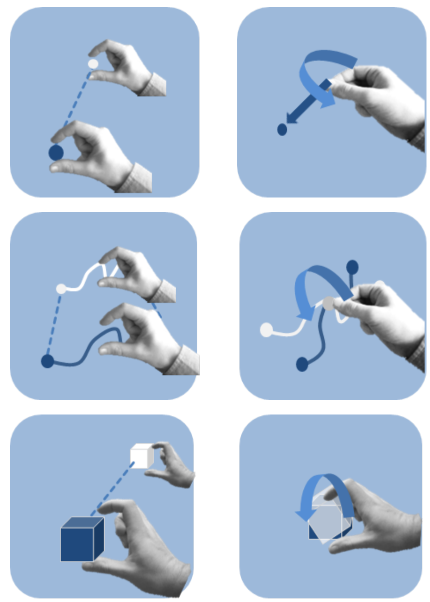

Fig. 5. Exemplary manipulation of virtual objects using gestures including translation and rotation for poses (top), trajectories (center) and tasks (bottom).

IV. METHODOLOGIES

A. Spatial Program Adaption

To implement the proposed spatial programming approach our system consists of the following basic hardware components: industrial robot, handheld device and motion tracking system (Fig. 6). The application of a motion tracking system enables both: tracking of human hands and objects. As a result, the tracking of objects could be used for pose tracking of the handheld device to provide markerless AR (see section 5). We use TCP/IP sockets for communication between the components of the system. A unified control layer for vendor independent control and programming of industrial robots via arbitrary input devices was previously presented in [11]. This control layer is also used in this project for program representation, execution and adaption.

Within our application, it is necessary to transform pose information between different coordinate systems, e.g. from

Fig. 6. Basic components of the spatial interaction system.

motion tracking system to robot. Fig. 7 illustrates the most important coordinate systems and some examples for related homogenous transformations. A calibration pose for the handheld is needed to start motion tracking and to calibrate initial orientation of the handheld in case of markerless AR. Furthermore, unknown transformations, e.g. between robot and motion tracking system, can be determined with the help of a calibration pose.

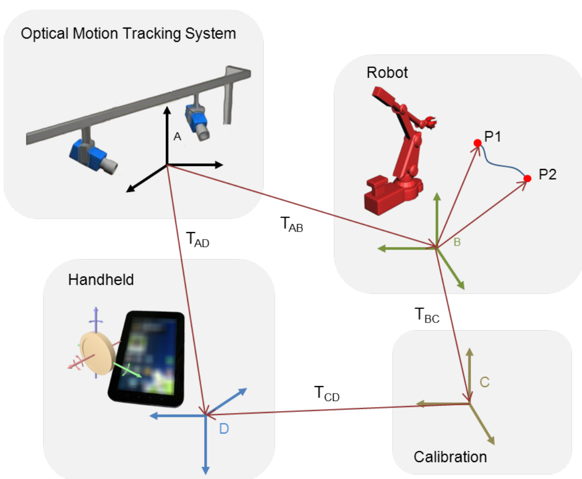

Fig. 7. Relevant coordinate systems for the transformation of pose information.

B. Mobile Augmented Reality

Prerequisite for a spatially adequate visualization of poses, coordinate systems, trajectories and further information in AR applications is the pose information of the camera. This means translational and rotational displacement between the camera coordinate system and a reference coordinate system must be known. There are various approaches using different sensor concepts for the determination of pose information for AR on handheld devices. Outdoor applications often use GPS in addition to internal rotation determination. Another approach uses fiducial markers with known dimensions. The markers are recognized through 2D pattern recognition including pose estimation. Based on the marker position in the 2D camera image and its known dimensions, the algorithm determines the pose information of the handheld in the marker coordinate system. Several frameworks and popular applications, e.g. ARToolKit, follow this approach. In order to improve pose accuracy, one can combine data from visual marker detection with the internal sensor data of the handheld [18], [19]. Besides a camera and a magnetic sensor, modern handhelds are equipped with a 3-axis gyroscope and a 3-axis acceleration sensor. These inertial sensors are subject to significant noise disturbance and other error interferences. However, these can be compensated partially through sensor calibration methods. Nevertheless, the combination of visual pose estimation with internal sensor data can therefore serve adequate accuracies for mobile AR applications [20], [21].
We aim for a markerless solution to be more flexible while moving in the robot environment. Hence, the objective is to use a multi-camera motion tracking system (prospectively), inertial sensors, sensor fusion and adequate calibration methods. In section 5, we present a solution for the determination of the handheld pose through sensor fusion using an extended Kalman filter.

C. Gesture Recognition

Gesture recognition for spatial interaction with virtual objects can be put into practice via 3D as well as 2D motion tracking. Enabling adequate 3D interaction based on 2D images from a single camera works only under fixed constraints. Otherwise, the algorithms are inaccurate because of the missing depth information. However, a rough determination of 3D movements for hands with known dimensions still is possible, e.g. see [16].
Due to the fact that finger gesture recognition based on 3D optical motion tracking data is very complex (see. application for MS Kinect [22]), we choose a novel approach to provide generic gestural interaction. For the manipulation of the virtual objects in AR we combine 2D gestures, recognized through the camera image of the handheld as command gestures, with 3D hand trajectories, tracked by the external motion tracking system.
Fig. 8 illustrates the flow of information combining 2D command gesture recognition with 3D motion tracking. The reasoning and processing unit provides feedback about the gestural manipulation via AR (visual) and vibration of the handheld device (haptic). Finally, it adapts the robot program according to the gestural manipulation. In the following, we give a closer insight into finger gesture recognition. The recognition of finger gestures contains the segmentation of skin color region, extraction of fingertips as features and a shape based pattern classification. Segmentation is done through skin color tracking. The main challenge is to make the application robust to different skin colors and lighting variations. This is a difficult task on the basis of limited computational effort and poor camera parameter handling on the handheld: e.g. it is not possible to completely turn off brightness and color control. As a result, we follow a efficient

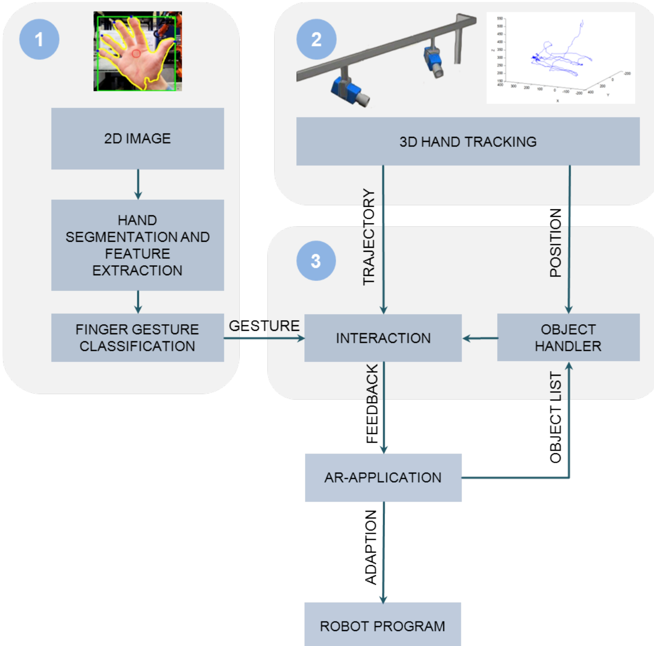

Fig. 8. Interaction based on gestures and AR consisting of 1) 2D gesture recognition using the camera of the handheld, 2) tracking of 3D trajectories of the hand tracked by a motion tracking system and 3) reasoning and processing unit.

probability-based approach for robust and fast skin color detection, presented in [23]. First of all, we convert the image to HSV color space ignoring the V channel. For a sample skin image, we determine a histogram, which is used as a calibration model for skin color. For segmentation, we finally compute backprojection, i.e. the probability that a pixel has skin color, threshold and smooth detected skin regions. For the purpose of features we consider the fingertips, which are determined through convex hull according to the principle in [24]. For the classification of gestures through the trajectories of single fingertips, we implemented an algorithm for 2D shape analysis, we consider Procrustes analysis [25]. The algorithm compares a trajectory with a reference trajectory by translation, uniform scaling, rotation and finally shape comparison. Therefore, we are able to robustly detect snap and release gestures based on the trajectories of the fingertips.

V. EXPERIMENTS

In this section, we present a low-cost solution of spatial programming, show results and discuss further work. We developed an Android App for the Samsung Galaxy S II smartphone, enabling an AR application based on OpenGLES. For 2D gesture recognition, we use OpenCV 2.4.0 Java bindings for the Android platform. Hand tracking and object recognition is carried out by Kinect Sensor and OpenNi Framework. The objective of the experiment is to prove the concept of spatial programming. The experiment covers the programming steps definition, evaluation and gestural adaption of poses. Poses are defined sequentially by pointing gestures and connected through linear movements. Synchronously, the trajectory is displayed in AR on the handheld device. Afterwards we manipulate the positions of single poses by snap gestures, displacement and release gestures. Finally the program is transferred from the handheld device directly to a KUKA robot controller. Also the program execution is done via the handheld device. For the experimantal setting we still do not consider a complex tracking algorithm of the handheld enabling markerless AR. In addition to a simple edge based object detecting algorithm using the optical motion tracking system according to [26], a sensor fusion with inertial sensors (accelerometers and gyroscopes) is already implemented on the handheld. To minimize errors affected by inertial sensors we implement calibration methods to determine bias and scale factor errors. The data fusion using an extended Kalman filter is implemented based on basic fusion principle presented in [18], [19], [20], [21]. In contrast to these publications, the magnetic sensor of the handheld is used for the determination of orientation. This is done to achieve more precise orientation information when the device is static in comparison to orientation determination by 3D tracking. The orientation takes into account the calibration pose presented in Fig. 7. Fig. 9 illustrates the structure of the extended Kalman filter with the objective of pose estimation including prediction and correction steps.

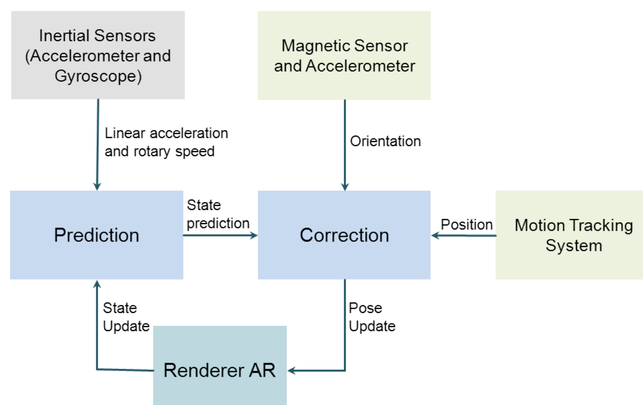

Fig. 9. Structure of markerless handheld pose estimation based on vision data of motion tracking system, inertial as well as magnetic sensor data.

A. Results and Discussion

Fig. 10 illustrates the skin color segmentation and the feature extraction. Fig. 11 shows a use case defining poses by pointing gestures and manipulating poses in AR by spatial interaction. A video of the experiment can be obtained at http://www.youtube.com/watch?v=Re6-xUKWUDE. Compared to conventional programming techniques, we achieve a significant reduction of programming time. This is to be analysed in more detail for simple and complex programming tasks. Currently, we achieve a frame rate of about 5fps for a resized image resolution of 400x240, when AR and gesture recognition are enabled. Nevertheless, it is possible to increase the frame rate through code optimization and the use of C++ OpenCV. Regarding

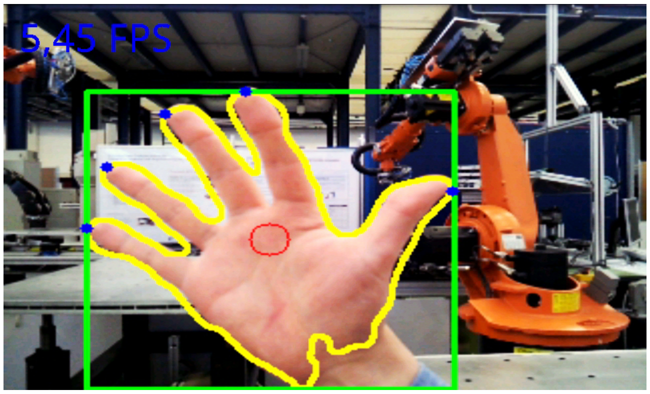

Fig. 10. Skin color segmentation and feature extraction.

pose tracking of the handheld, we identified the use of a magnetic sensor for orientation determination as a bottleneck of the current system. This sensor is liable to provide errors. Further considerations will take into account an improvement of orientation estimation based on visual object tracking. Initial project results (see Fig. 12) show accuracies of the handheld recognition for a simple trajectory. The use of the Kalman filter increased the obtained standard deviation from about 70 mm to 50 mm. As a reference system we used the precise Motion Capture Systems Vicon MX within our experiments. Further work will compare different object tracking principles and filtering methods as well as resulting tracking accuracy and visualization errors in AR. Thus, we aim to identify an adequate sensor concept for general applicability. Furthermore novel gestures and interaction methods for spatial programming will be examined. Another issue is the possible occlusion of virtual objects in the AR by real objects. In order to illustrate the current state of our research, we give an outlook about ongoing work:

- enhance performance of 2D gesture recognition algorithm,
- find adequate pattern recognition and gestures for interaction on task level,
- implement a gestural interaction method for the manipulation of the orientation,
- implement markerless tracking of handheld device,
- consider possible occlusions in AR application,
- implement interfaces for program exchange between App and Digital Factory software tools,
- evalutate sensor concept and make adaptions, due to adequate sensor innovations,
- performance tests and evaluation toward more applications.

Besides the intutive interaction, there is a disadvantage of our system in comparison to programming methods using haptic interaction between industrial robot and human, e.g. Lead-Through-Programming. Toward the user performing a gripping task with virtual objects without haptic feedback is quite unfamiliar. Using the vibration of the handheld is a minor substituation. However the vibration helps the user to get used to the interaction method.

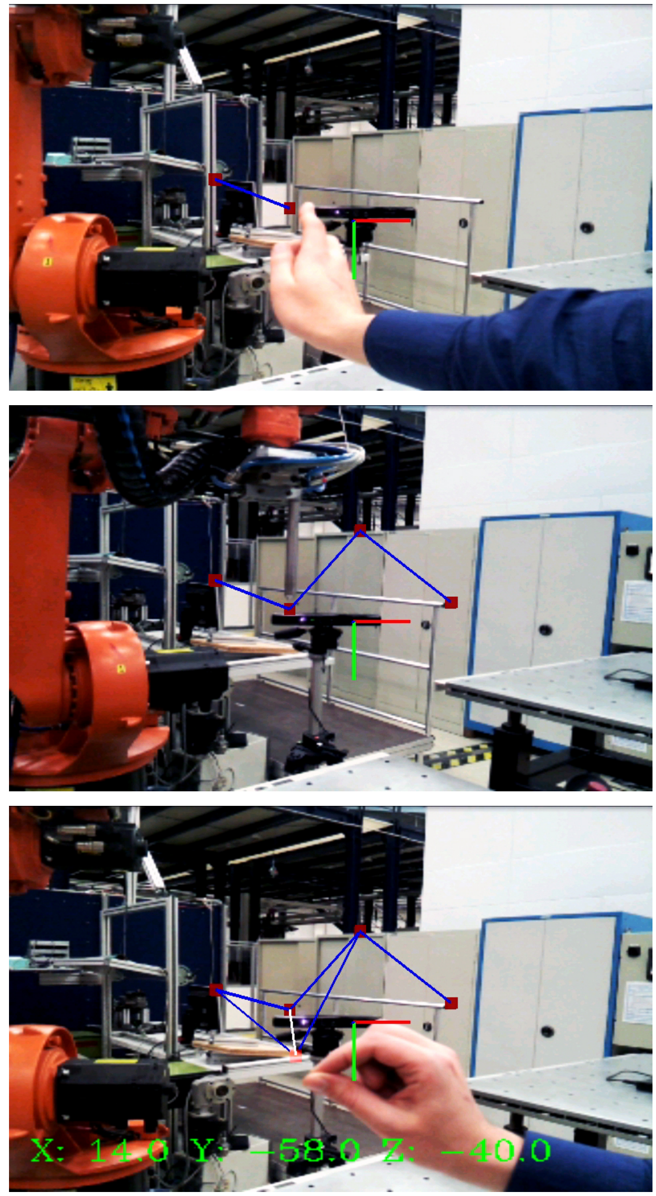

Fig. 11. Pose definition (top), program execution (center) and pose adaption (bottom).

VI. CONCLUSIONS

The overall potential of spatial programming lies in a reduction of programming times and an ease of use. By means of spatial robot programming, the programmer is supported by the help of a highly efficient assistance system for online programming. The intuitive usage of spatial interaction represents a major simplification for industrial robot programming compared to conventional online methods. Our system enables non-specialists to define, evaluate and manipulate robot programs without broad practice and experience. The presented method can also be used in combination with conventional methods of robot programming.

The possible fields of applications for definition, evaluation and manipulation of robot programs seem numerous.

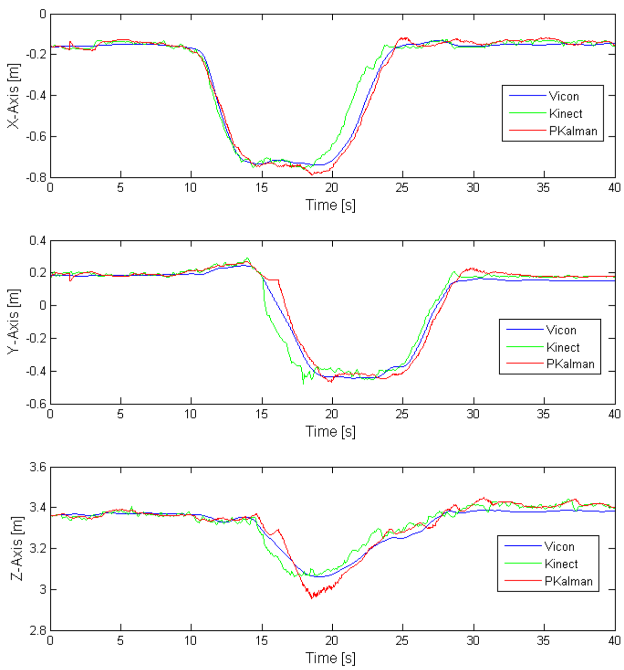

Fig. 12. Positional deviation for a sample trajectory of the handheld device: reference system Vicon MX (blue), Kinect (green), sensor fusion (red).

However, depending on the specific application, the usage of a single module could be more or less meaningful. The gesture based definition of complex paths could be particularly interesting regarding spray painting tasks. AR based spatial evaluation of programs on ubiquitous handheld devices could potentially fit nearly any robotic application. Among the definition of Pick'n'Place tasks, a gesture-based definition of more complex tasks demands additional sensors to achieve a required accuracy, which still cannot be reached by most markerless motion tracking systems and algorithms. Further limits of the applicability are caused by the complexity of special robot tasks, e.g. in-process path planning for dynamic environments.

REFERENCES

[1] J. Krüger, R. Bernhard, D. Surdilovic and G. Spur, Intelligent Assist Systems for flexible Assembly, Proceedings of the 56th CIRP General Assembly, Kobe, Japan, pp. 29 - 32, 2006.

[2] S. Forge, C. Blackman, A Helping Hand for Europe: The Competitive Outlook for the EU Robotics Industry. EUR - Scientific and Technical Research Reports, Publications Office of the European Union, 2010.

[3] R. Dillmann, M. Zöllner, M. Ehrenmann and O.Rogalla Interactive Natural Programming of Robots: Introductory Overview, 2nd IARP/IEEE-RAS Joint Workshop on Technical Challenge for Dependable Robots in Human Environments, 2002.

[4] T. Stiedl, Multimodale Interaktion mit Automatisierungssystemen, Dissertation, Universitt Stuttgart, Shaker Verlag, Aachen, 2010.

[5] A. Kazi, J. Bunsendal, D. Haag, R. Baum and R. Bischoff, Next Generation Teach Pendants for Industrial Robots, Advances in Human-Robot Interaction, Star 14, Springer-Verlag Berlin Heidelberg, pp. 47-66, 2005.

[6] B. Akan, A, Ameri, A., B. Cürüklü and L. Asplund, Intuitive Industrial Robot Programming Through Incremental Multimodal Language and Augmented Reality, IEEE International Conference on Robotics and Automation, pp. 3934-3939, Shanghai, 2011.

[7] P. Neto, J.N. Pires and A.P. Moreira, Accelerometer-based control of an industrial robotic arm, The 18th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 2009), pp. 1192-1197, Toyama, 2009.

[8] K. Dixon, S. Iba, J. Dolan and P. Khosla, Gesture-based Programming for Arc Welding, in Proceedings of the International Conf. on Intelligent Robots and Systems (IROS) 2003, pp. 3479-3484, Las Vegas, 2003.

[9] A. Ameri E., B. Akan and B. Cürüklü, Augmented Reality Meets Industry: Interactive Robot programming, Proceedings of SIGRAD 2010: Content aggregation and visualization, Västeras, pp. 55-58, Sweden, 2010.

[10] J. Lambrecht, M. Kleinsorge and J. Krüger, Markerless Gesture-Based Motion Control and Programming of Industrial Robots, 16th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2011), Toulouse, 2011.

[11] J. Lambrecht, M. Chemnitz and J. Krüger, Control layer for multi-vendor industrial robot interaction providing integration of supervisory process control and multifunctional control units, IEEE Conference on Technologies for Practical Robot Applications (TePRA), pp. 115-120, Woburn, 2011.

[12] S. Hussein, M. Buechel and J. Krueger, Stable, adaptive interaction and contact transition control of a high inertia haptic interface for haptic simulation in gait rehabilitation, Proceeding of the 2011 IEEE International Conference on Robotics and Automation (ICRA 2011), 2011.

[13] A. Postawa, M. Kleinsorge, J. Krüger and G. Seliger, Automated Image Based Recognition of Manual Work Steps in the Remanufacturing of Alternators, Advances in Sustainable Manufacturing: Proceedings of the 8th Global Conference on Sustainable Manufacturing, Springer Berlin Heidelberg, pp. 209-214, 2010.

[14] S.K. Ong, M.L. Yuan and A.Y.C. Nee, Augmented reality applications in manufacturing: a survey, International Journal of Production Research, 46:10, pp. 2707-2742, 2008.

[15] G. Reinhart, U. Munzert and W. Vogl, A programming system for robot-based remote-laser-welding with conventional optics, CIRP Annals - Manufacturing Technology, Vol. 57, No. 1, pp. 37-40, 2008.

[16] S.K. Ong and Z.B. Wang , Augmented Assembly Technologies Based on 3D Bare-hand Interaction, Annals of CIRP, Vol. 60, No. 1, pp. 1-4, 2011.

[17] H.C. Fang, S.K. Ong, and A.Y.C. Nee, Robot Programming using Augmented Reality, International Conference on CyberWorlds, pp. 13-20, 2009.

[18] S.G. Chroust and M. Vincze, Fusion of Vision and Inertial Data for Motion and Structure Estimation, Journal of Robotic Systems 21 (2), Wiley Periodicals Inc., pp. 73-84, 2004.

[19] Y. Tao, H. Hu and H. Zhou, Integration of Vision and Inertial Sensors for 3D Arm motion Tracking in Home-based rehabilitation, The International Journal of Robotics Research, Vol. 26, No. 6, SAGE Publications, pp. 607-624, 2007.

[20] S. You and U. Neumann, Fusion of Vision and Gyro Tracking for Robust Augmented Reality Registration, Proceedings. IEEE Virtual Reality, pp. 71-78, 2001.

[21] J.D. Hol, T.B. Schon, F. Gustafsson and P.J. Slycke, Sensor Fusion for Augmented Reality, 9th International Conference on Information Fusion, pp. 1-6, Florence, 2006.

[22] I. Oikonomidis, N. Kyriazis and A.A. Argyros, Efficient model-based 3D tracking of hand articulations using Kinect, in Proceedings of the 22nd British Machine Vision Conference, BMVC 2011, University of Dundee, UK, 2011.

[23] G.R. Bradski, Real Time Face and Object Tracking as a Component of a Perceptual User Interface, Fourth IEEE Workshop on Applications of Computer Vision (WACV'98), pp. 214-219, 1998.

[24] D. Lee and S.G. Lee, Vision-Based Finger Action Recognition by Angle Detection and Contour Analysis, ETRI Journal, Vol. 33, No.3, pp. 415-422, 2011.

[25] M. Wilhelm, D. Roscher, M. Blumendorf and S. Albayrak, A Trajectory-Based Approach for Device Independent Gesture Recognition in Multimodal User Interfaces, Lecture Notes in Computer Science, Vol. 6306/2010, pp. 197-206, 2010.

[26] J.P. Lima, V. Teichrieb, J. Kelner and R.W. Lindeman, Standalone edge-based markerless tracking of fully 3-dimensional objects for handheld augmented reality, VRST 09 Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology, ACM New York, pp. 139-142, 2009.