Robot Manipulator Programming Interface based on Augmened Reality

Alexander Schwandt, Arkady Yuschenko

Abstract*:- *The integration cost of industrial robots into small and medium sized enterprises (SMEs) is nowadays one of the main obstacles to make automated solutions profitable. The standard implemented human robot interface (HRI) in industrial robots, is complicated and time consuming even for trained technicians. A well-known trend in robotic manipulator development, known in the field of intelligent HRI, is aimed at to make automated solutions with industrial robots cost-effective. A HRI based on augmented reality (AR) is presented to cut short the programming phase during commissioning of industrial robots in the production line. The system proposed in the given article was implemented on a portable computer in accordance with an AR-Stylus using safety goggles mounted with a high-resolution camera and a wearable display. The hardware setup presented was used in an experiment to show, how the HRI's basic functions, namely intuitive robot programming by demonstration (PbD), visualisation and simulation of industrial robots program, could be implemented. A high level of encapsulating the programmer into an AR environment was achieved by configuring a camera tracking the 6D position of the AR-Stylus and a real-time projection of the 3D robot gripper model virtually substituting the AR-Stylus. In contrast to the HRI based on the conventional PbD method, this paper covers a new type of PbD called “virtual robot programming by demonstration” (vPbD). The accuracy of the HRI system in determining the position of the AR-Stylus was experimentally analysed. In a sequential robotic program, apart from visualising the endpoints and the trajectories in WD, a real scale 3D model projection of the robot gripper is aligned in real-world environment from the observer's perspective. This enables the operator to detect potential collisions of the robot gripper with the surrounding with efficacy. Further conclusions and developments pertaining to the study will be followed.

Keywords: Robot Manipulator • Human Robot Interface • Augmented Reality • Robot Programming by Demonstration • Collaborative Robotics • Intuitive Programming.

I. INTRODUCTION

Similar to automobiles, industrial robots are nowadays common in mass production industries. Furthermore in small and medium sized enterprises (SMEs), the demand for industrial robots have increased in the last few decades. Yet there is no broad spectrum exposure of industrial robots in this sector. This can be owed to the fact that industrial robots pose high costs of integration and (re)programming during an ongoing production, which might change frequently due to small scale productions. In order to make industrial robots more common within SME, the human robot interface (HRI) should be made systematic, simple and self-explanatory. This renders an affordable solution for SME demands to automate production steps in frequently changing production lines. As a result no professional adjustment or reprogramming is required.

Normally the programming procedure for an industrial robot is sequentially setting the tool center point (TCP) at the desired points, subsequently saving them in a program. Unless and otherwise tutored and used widely, this procedure poses several safety risks for the operator. In the traditional programming initially the robots are moved using the teach pendant which is often inconvenient in getting the TCP to the desired point, where the operator's focus changes frequently between the robot and the teach pendant, thus slowing down the entire process of programming. Moreover the operator may surpass the robot’s moves. After the program points have been taught to the robot, the resulting program can only be visualized in a simulation environment. Usually during in commissioning, the operator evaluates the robot's program gradually. Repeating the above mentioned method several times can be time consuming and complicated for amateur trainers programming a simple operation for the robot. An approach to accelerate the programming phase is done by using a portative system. This system provides the operator with the needed information and visualization in terms of augmented reality (AR) through a 6dofwearable display, and receives simple high level teaching commands via buttons on an ergonomic AR-Stylus.

B.Akan proposed a system in which AR is combined with Virtual Reality (VR) environment and a speech recognition system to implement an intuitive HRI. AR stands for computer generated graphics, text and three dimensional (3D) models, which overlay on real video stream. In comparison to B. Akan’s system, the system presented differs in the entire visualization and simulation tasks where the operator can move freely. Thanks to a wearable display, the operator’s view in an AR environment is free of interference. This feature allows the observer to maximally integrate into the AR environment. Moreover no additional sensor equipment’s, like cameras are needed to be installed on a robot. For determining the observer’s perspective, it is necessary to install planar QR-markers on the end effector.

The main aim of this approach is to integrate additional information into the operator’s real visual field. he 3D optical visualisation of processes in computer numerically controlled (CNC) machines, as described by Olwal [3], serves as an

Revised Version Manuscript Received on 16 September, 2019.

Alexander Schwandt, Robotic Center of Education and Research, Bauman Moscow State Technical University, Moscow, Russia Izmaylovskaya , Moscow, Russian Federation.
(Email: *:* shvandta@student.bmstu.ru)

Arkady Yuschenko, Professor Robotic Center of Education and Research, Bauman Moscow State Technical University, Moscow, Russia Izmaylovskaya Sq., 7, 105007 Moscow, Russian Federation.
(Email: arkadyus@mail.ru)

example for the usefullness of AR in industrial applications. AR proves to be advantageous in the field of human robot interaction, where especially in tele-robotics applications, AR allows the operator to perceive things as if he were present at the remote target environment [4-6].

In addition a portative system and a wearable display was used in paint applications for convenience. Reference points were projected into the operator‘s field of view on to the working piece, supporting the operator for orientation[1]. A similarily successful AR application was proposed by Po Chen Wu [7], tracking a pen with calibrated dodeca shaped head with planar QR-Markers in 6Dof . This function is not further discussed in this paper and is only used for defining points during the programming with the AR-stylus.

In this paper the AR environment refers to the output of the AR-Module and serves as the main work environment for the operator. In order to analyse the robot's program, the three use cases highlighted and taken into consideration are as follows : visualisation of robot‘s sequential program, virtual motion simulation of the robot during program execution, and programming procedure of the industrial robot using the AR-Stylus.

II. SYSTEM ARCHITECTURE

The hardware components in the proposed system are a 6DoF KUKA KR10 manipulator, a pneumatic gripper Schunk, a robot controller KR C4 compact and a portable single board computer phyBOARD-SeginARM CortexTM-A7. The hardware used for AR as shown in Fig. 1 are as follows : a wearable display *Vufine+* mounted with 3MP resolution camera on safety goggles (A), a teaching stylus with remote control buttons (B) and an attachment for camera recognition as shown with a gripper model (C).

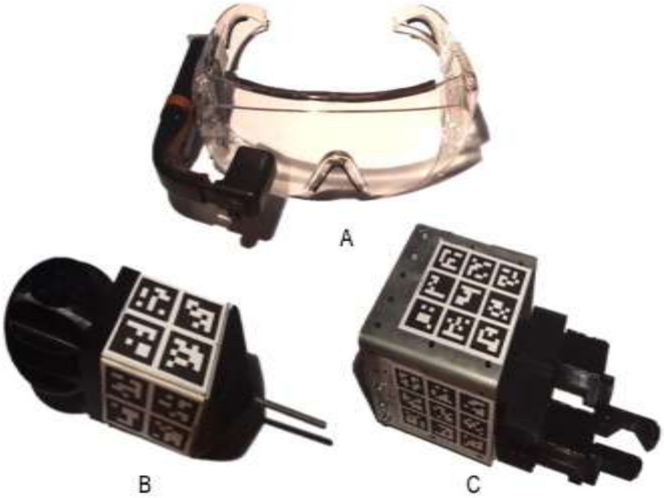

Fig. 1. AR hardware of the proposed system.

The AR-Module acts as a front to the user by projecting information into the operator‘s view within the wearable display. The architecture of the system presented in Fig. 2 is described in detail in the following paragraph. Using the over-head mounted camera, the operator‘s perspective is determined by using an attachment with the planar QR-Codes mounted on the robot‘s flange. This attachment is further taken as a reference coordinate system for AR projections. The position of the robot‘s flange is frequently transmitted by the controller to the reasoning system and forwarded to the AR-Module. With the help of the AR-Stylus and the remote control buttons on it, the operator programs 6D points to the robot. These points are defined by a camera recognised AR-Stylus position and orientation and is transformed into robots world coordinate system.

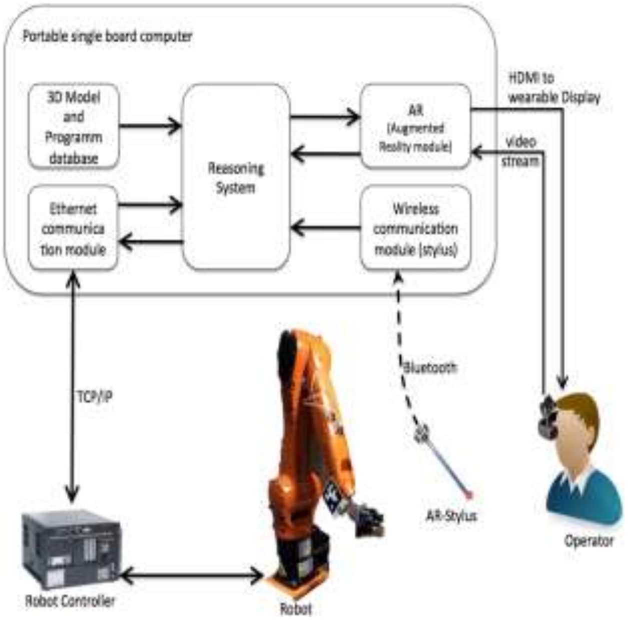
Portable single board computer
3D Model and Programm database
Reasoning System
AR (Augmented Reality module)
Ethernet communica tion module
Wireless communication module (stylus)
HDMI to wearable Display
video stream
Bluetooth
TCP/IP
AR-Stylus
Operator
Robot Controller
Robot

Fig. 2. Hardware and software components of the system.

III. AUGMENTED REALITY ENVIRONMENT & RESULTS

The AR environment consists of a data base, containing all virtual models, e.g. robot’s grippers, end effectors and work objects. Furthermore the database contains a backup of robots teached points and trajectories, represented in the robot’s world coordinate system. Thanks to this representation, the object positions can be easily transformed between coordinate systems (CS), as well into to operators perspective and field of view.

The cameras video stream is reprojected by the AR-Module into wearable display in, that observers field of view is not disrupted by any means. Since the wearable display doesn’t covers the entire operator’s view, only in this view field area the high informative AR environment is visible. In the respective approach the determined operators perspective is taken as key reference system for further visualisation and simulation tasks in AR-Module (see *CScamera* in Fig. 3). An important advantage is, that only additional 3D objects, points and trajectories need to be added to observer’s view, instead of simulating the entire environment. This fact can be considered as an advantage, because less processing performance is required for visualisation tasks, as if the entire environment had to be simulated in case of virtual reality. The illustration of Fig. 3 presents an overview of the systems coordinate transformations and coordinate systems. The homogeneous transformation matrices T_{MARKER},T_{TOOL},$T_{STYLUS\text{-}TIP}$ are generally known in the context of initial calibration procedures [8], nevertheless not further considered in this paper. The transformation matrix T_{FLANGE} is given by robot’s controller and T_{CAMERA} and T_{STYLUS} are determined by the head mounted camera, as long QR-Markers are recognised. With the help of the in the paper presented hardware setting a reliable pose estimation for T_{CAMERA} and T_{STYLUS} was experimentally estimated in a distance between camera to QR-Marker from 0.3 m up to 1.5 m with an measurement accuracy of linear coordinates $\pm 1.5 \bullet 10^{-1}$mm, and angular co-ordinates $\pm 6 \bullet 10^{-2}$*degree.* In the following demonstration scenarios a industrial robot KUKA KR10 R1100 is presented, taking with a conical shaped gripper unmachined pressure casted work pieces off a belt.

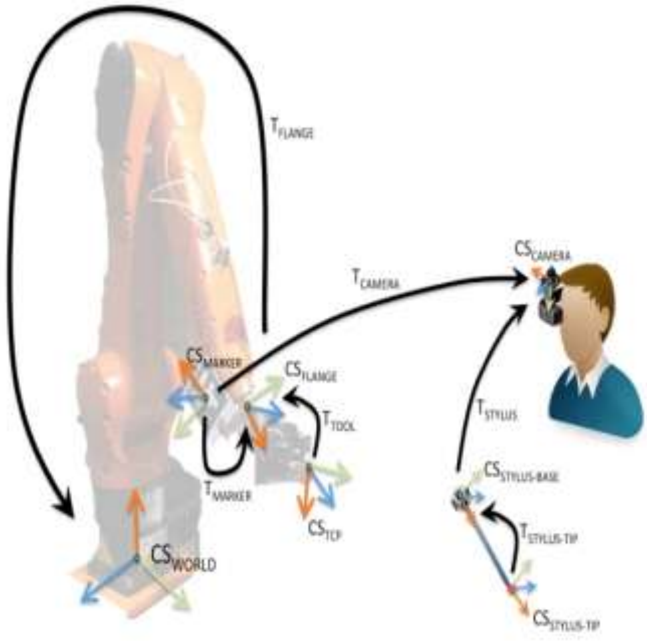

Fig. 3. Overview about homogeneous coordinate transformations in proposed system.

Since the operator's perspective is determined in AR environment, the operator can freely move and observe the target scene from different perspectives for better understanding. With a 3D model of end effector, in this case of grip- per, a real live scaled projection of the model can be embedded into operators view. For simplified simulation proposes, the model could be backed up in sev¬eral conditions, that the appropriate can be chosen. In this meaning a gripping process with moving gripper fingers could be shown in discrete steps. More over a collision evaluation of end effector, moving along the programmed trajectory can be done, observing the simulated model from various perspectives in work AR environment.

The usecase, visualisation of robot program points, will be in the following described. In Fig. 4 is shown, how a visualisation of sequential robot's programs could look like in operator's perspective in AR environment. Points of the selected program are highlighted in color and marked with point name, while points of unselected program's can be displayed as additional information in background colours. This allows to evaluate, how the trajectory would look like in the real work environment, without moving the robot, as in above mentioned traditional procedures of programming industrial robots.

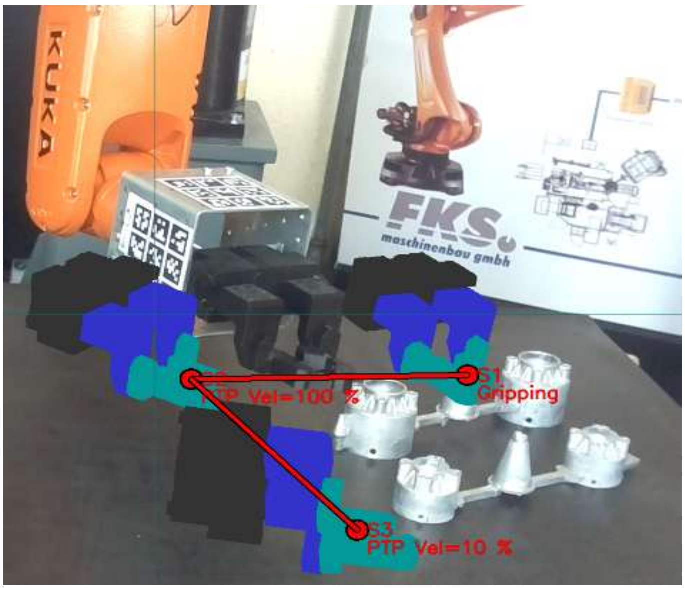

Fig. 4. Visualisation of robot program points.

According to Fig. 3 the trajectory point, represented in robot's world coordinate system, are homogeneous transformed into the recognised marker's coordinate system CS_{MARKER} as follows:

$$S1_{MARKER} = T_{MARKER}{}^{-1} \cdot T_{FLANGE}{}^{-1} \cdot S1_{WORLD} \qquad (1)$$

were T_{FLANGE} is given by the robot controller, T_{MARKER} is known in the context of initial calibration in relation to CS_{FLANGE} . T_{CAMERA} is measured by head mounted camera and is called further on, neglecting the fourth row, camera homography matrix H. As example the program point $S1_{WORLD}$ is taken, represented in robot's CS_{WORLD}. It is well known, that the program point S1, represented in CS_{MARKER} of recognised marker as a column vector, can be projected by the homography matrix and other transformations C_{Opt}, describing camera's optical reprojections, into the two dimensional (2D) image plane of operator's wearable display as follows, see Fig. 5:

$$S1_{IMAGE} = C_{Opt} \cdot H \cdot S1_{CAM} \qquad (2)$$

where the coordinates of the projected point in 2D image plane are represented as follows:

$$S1_{IMAGE} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \qquad (3)$$

The earlier mentioned second use case, simulation of robot motion during executon of sequential program, will be described below.During programming phase of industrial robots, great attention is given to trajectory evaluation in order to achieve collision free robot movement and effective program performance. Programming the robot in offline mode, in simulation tools the entire robot movements can be analysed. In order to achieve accurate simulation results, a detailed model of robot cell must be implemented in simulation environment. As above mentioned, in presented approach, realised only on AR, only additional infomations are embedded into operators view. With integration of 3D models of robot's end effector's or other relevant objects into operators environment, different simulation features can be realised.

Since the real scale model of robot’s end effector can be projected at every point of trajectory in AR environment, the operator is able to make visual collision checks with surrounding environment objects. Moreover, introducing a time referenced scheduler for simulating end effector motion along the trajectory points, a real scale movement could be shown in AR environment. For improved simulation quality a dynamic model of the manipulator and a inter point motion simulation should be implemented. In demonstration scenario was shown, how trajectory visualisation during collision check could look like in AR environment. The robot’s program consists of 3 points in order to grab the work piece with the conical shaped gripper. In Fig. 6 is the operators view presened, showing a AR overlay in wearable display with the robot’s gripper at every of the programmed points as a virtual real scale 3D model. To ensure, no collision of the gripper with surrounding objects would occur during it’s movement along the programmed trajectory, the operator can visually evaluate each of the situation from different perspectives.

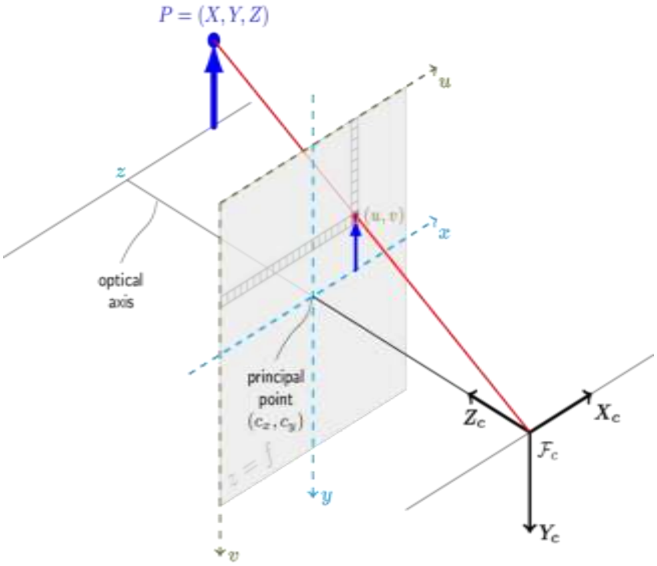

Fig. 5. Illustration of the pinhole camera model. [9]

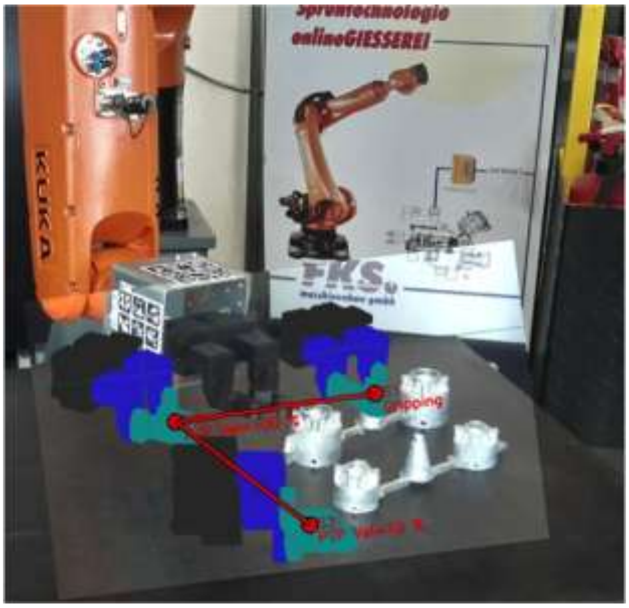

Fig. 6. Visualisation and simulation of robot motion during program execution.

3.1 Intuitive programming procedure of industrial robot with AR-Sytlus.

A method is needed to allow the operator to define the position and the orientation of the robot’s tool with ease, in order to speed up the programming phase of industrial robots. The proposed approach, using a AR-Stylus with buttons which is 6D-tracked by the operator‘s head mounted camera, could reshape the programming phase more convenient when compared to using a teach pendant. By pointing the AR-Stylus in the work space of the robot, the position and the orientation can initially be taught to the robot, by pressing one of the buttons. By pressing an AR-Stylus button, the estimated 6D point of the AR-Stylus tip, shown in Fig. 3 as $CS_{STYLUS\text{-}TIP}$, will be transformed by the AR-Module into the robot’s CS_{WORLD}, according to the following equation:

$$S1_{MARKER} = T_{FLANGE} \cdot T_{MARKER} \cdot T_{CAMERA}{}^{-1} \cdot T_{STYLUS} \cdot T_{STYLUS-TIP}\ , \qquad (1)$$

The AR-Stylus is connected with a portable computer via a wireless connection and is equipped with five buttons which can be preconfigured for various types of programming points. Once a button signal is received, the measured and the assigned point type is sent into reasoning system for reachability checks. If the checks prove to be successful, the point will be added to the robot’s program.

In order to increase the user-friendly aspect of the programming interface, the real scale 3D model of the end effector can be projected in place of the AR-Stylus. This provides an effect where the operator holds the robot’s tool, which in reality could be humanely impossible to lift, in his hands. Manually guiding the robot by the endefector is actually similar to the programming by demonstartion (PbD) method. In the proposed approach the real robot is involved only as a reference CS for the AR environment and the robot’s controller can be switched off. This programming method can be called „virtual programming by demonstration“ (vPbD) because a virtual robot is manually guided by the operator. Programming a usual industrial robot by vPbD like a collaborative robot increases the operator‘s safety in a way and requires no relevant modifications of the robot.

The operator‘s view in the AR environment with the projected 3D gripper model on AR-Stylus is shown in Fig.7. The projection of the gripper displayed in the AR environment would completely correlate to the real gripper, as though this point was programmed and perceived by the real robot. This feature demands specific orientation skills from the operator and needs to be further investigated in regards to the mental capabilities as proposed by Yuschenko A.S. [10].

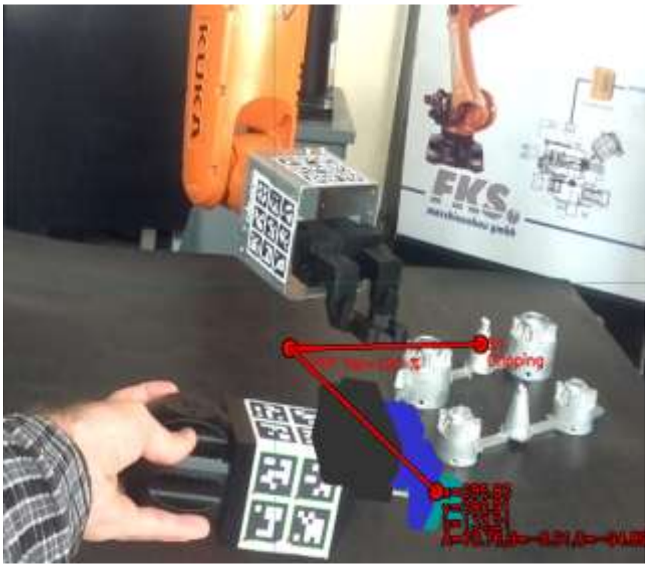

Fig. 7. Programing points with AR-Stylus for sequential program to robot.

IV. CONCLUSION

The system proposed represents an alternative HRI, integrating the operator markedly into the AR environment by using a camera tracking the 6D position of the AR-Stylus and a real-time projection of the 3D robot gripper model virtually substituting the AR-Stylus.

Commerical methods of programming industrial robots are time consuming whereas the AR-HRI is time conserving as well as cost effective.

This presentation carries a few drawbacks that need to be further studied, for example, the accuracy of the visual output from the AR-Module should be thoroughly analysed. In addition to the mentioned simulation capabilities in the AR environment such as the visual check for trajectory collisions, the entire robot structure should be considered for practical purpose. This proposal which estimates the position of the planar QR-Markers by a single camera, can lead to a lack of accurate 3D projections in situations like the AR-Stylus being partly hidden and the 3D model of the end effector being projected on top of an obstacle. To overcome this shortcoming, a time of flight camera and further algorithms, processing 3D image data should be used. In order to increase the simulation of a robot in the AR environment, a physical model describing the robot’s dynamics shall be used for the robot’s motion.

V. ACKNOWLEDGEMENT

This work is financially supported by RFBR, project No.18- 07-0131. I would also like to extend my gratitude to FKS-Maschinenbau GmbH for their support in making this venture possible.

REFERENCES

1. Petterson, T., Pretlove, J., Scorub, C., Engedal, T.,Lokstad, T.: Augemted reality for programming industrial robots. In: The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings, pp. 319-320, (2006)
2. Akan, B., Ameri, A., Curuklu, B., Asplund, L.: Intuitive Industrial Robot Programming Through Incremental Multimodal Language and Augmented Reality. In: IEEE International Conference on Robotics and Automation, (2011)
3. Olwall, A., Gustafsson, J., Lindfors, C.: Spacial augmented reality on industrial CNC-machines. In: Proceedings of SPIE, 6804:680409-680409-9, (2008)
4. Fang, H., Ong, S.K., Nee, A.Y.-C.: Robot Programming using Augmented Reality. In: IEEE, Sept.(2009).
5. Marin, R., Sanz, P., Sanchez, J.: A very high level interface to teleoperate a robot via Web including augmented reality. In: IEEE International Conference on Robotics and Automation(Cat. No.02CH37292). Proceedings, (2002)
6. Jara, C.A., Candelas, F.A., Gil, P., Fernandez, M., Torres, F.: An augmented reality interface for traning robotics through the web. Communication, pp. 189-194, (2005)
7. Po, C.-W., Wang, R., Kin, K., Twigg, C., Han., S., Yang, M.-H., Chien, S.-Y.: Accurate 6Dof Tracking of a Passive Stylus. In: The 30th Annual ACM Symposium on User Interface Software and Technology, Proceedings, pp. 365-374, (2017)
8. Siciliano, B., Khatib, O.: Springer Handbook of Robotics. Springer, Berlin Heidelberg (2008)
9. OpenCV 2.4.13.7 documentation page, https://docs.opencv.org/2.4. Last accessed 15 Apr 2019
10. Yuschenko, A.S.: Man and robot - ergonomic problems of collaborative robotics. In: World of psychology, 86-102 (2018)