

# Interactive Industrial Robot Programming for the Ceramic Industry

Regular Paper

Germano Veiga<sup>1,4,\*</sup>, Pedro Malaca<sup>1,3</sup> and Rui Cancela<sup>2</sup>

1 Departamento de Engenharia Mecânica, Universidade de Coimbra, Coimbra Portugal

2 Motofil Robotics, S.A., Aveiro, Portugal

3 Sarkis Robotics, Lda, Coimbra Portugal

4 INESC TEC, Porto, Portugal

\* Corresponding author E-mail: germano.veiga@inescporto.pt

Received 30 Aug 2012; Accepted 12 Jun 2013

DOI: 10.5772/56753

© 2013 Veiga et al.; licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

**Abstract** This paper presents an interactive programming method for programming industrial robots in ceramic applications. The main purpose was to develop a simple but flexible programming system that empowers the user with product driven programming without compromising flexibility. To achieve this flexibility, a two step hybrid programming model was designed: first the user sketches the desired trajectory in a spatial augmented reality programming table using the final product and then relies on an advanced 3D graphical system to tune the robot trajectory in the final workcell. The results measured by the end-user feedback show that a new level of flexibility was reached for this type of application.

**Keywords** Industrial Robot Programming, Augmented Reality, Human-Robot Interaction

## 1. Introduction

Robot programming is nowadays mainly done using online teaching pendant programming. This type of

programming is very well suited for industrial robots due to their high repeatability capabilities, but their use requires long setup times and jogging skilled operators. Despite these disadvantages it is the most common programming method for robots due to the intuitiveness, low initial cost and ease of mastering by low skilled workers, who are frequently specialists in the technological process rather than in robotics.

Significant research has been carried out in novel online programming techniques using advanced sensing and input techniques such as force-based guidance [1], vision-based guidance [2], the combination of force sensors and voice commands [3] [4], the combination of force sensors and vision systems [5], vision systems and CAD models [6], among others.

Offline programming is based on the CAD model and its acceptance has grown in recent years. Nevertheless, when compared with the CAM systems for machining applications, robot programming from CAD models is lagging behind. The users of offline programming systems have the advantage of working in the office and

have access to the automatic trajectory generation from the CAD models, but nevertheless, the amount of work to put the CAD workcell together, the required online adjustments and the required computer skills still limit offline programming usage.

A comprehensive review on robot programming techniques with a comparison of offline and online systems and an overview of research trends is provided in [7].

From the above it is clear that many industries would benefit from a balance between simplicity, intuitiveness and fast programming times. The ceramic industry, which is characterized by the low penetration of robots per plant, is certainly one of those cases. The importance of an advanced interface for supervision and parameterization is already recognized by the ceramic industry [8] but customizable programming techniques are still rare.

### 1.1 The ceramic tableware industry: an overview

Portugal has been a historic manufacturer of porcelain tableware for more than 100 years. Recently manufacturers needed to implement new forming technologies to meet the high quality demands of new worldwide clients. Since high-pressure casting, the traditional process, resulted in some quality issues many companies decided to use isostatic pressing systems, which produce ceramics of a better quality.

These systems not only assure product quality but also allow the production of more complex tableware shapes and higher production throughput, meaning that part finishing stages need to be more and more efficient to comply with production needs.

### 1.3 Industrial state of the art

The isostatically produced ceramic parts are dropped on to a conveyor device that transports the parts to an intermediate buffer.

The operator manually picks up the part and sets it on a support, driving it tangentially on to the edge of a grinding disk, eliminating flashes and excessive roughness, and leaving a smoother edge (Figure 1).



Figure 1. Manual grinding and fettling.

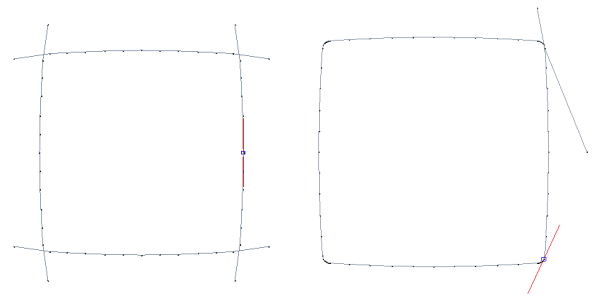


Figure 2. Different trajectories for the same ceramic for grinding on the left and fettling on the right.

Then the part is picked-up by another operator who fettles the part edges, ensuring a smooth finishing and a edge free contour (Figure 2). Fettling is the term normally given to the finishing of casted or injected parts to remove unwanted flashings, risers or similar material excess. In ceramics, fettling is normally a manual process done using hand-held soft tools, such as dry or wet sponges. Fettling normally follows the initial grinding stage.

There are currently completely automatic fettling systems on the market that rely on machine vision systems to acquire the part outline. These systems lack the flexibility of defining the custom trajectories that are needed in products with angular edges, especially for the grinding process.

### 1.2 Challenge

Manually programming the robot to finish complex and even asymmetrical ceramic parts complying with exact outside shapes would be an extremely time intensive task and very harsh for the end-user, who often has little to no programming experience.

Therefore, the challenge comes with the development of a robotic solution to finish ceramic tableware parts that not only ensures the compliance of all productive and quality requisites but also turns all programming and adjusting tasks into simple, intuitive and fast procedures.

One important requirement from the company was that trajectories should be fully customizable and not only dependent on the ceramic contour. This feature is very useful due to the flexibility that it gives to the programmer in more complex shapes. In Figure 3 different contour trajectories are applied to the same ceramic.

## 2. Methodology and background

Two main issues were addressed in this work: the development of a product oriented programming environment for the programming stage and an intuitive software application for adjustment and production control.

## 2.1 Augmented reality

Augmented reality (AR) consists of adding virtual information to a real scene. The combination of the visual information in real time uses one of the following three hardware platforms: (1) A Head Mounted Display or Optical See Through devices [9] project the current rendering of the virtual data on to a semi-transparent mirror. (2) Small handheld devices use a camera and a screen to provide video enhanced scenes. (3) Direct Augmentations use projectors and the surfaces of the environment [10].

Augmented reality use in research for robotics and industrial applications is not a novelty. For several years head mounted displays had been used in industrial applications [11] for the aerospace and automotive industry. In more recent years there has been a shift towards the use of video through devices and spatial augmented reality [12]. Different aspects of the topic have been addressed, namely collision avoidance [13] and teleoperation [14]. Nevertheless, the use of spatial augmented reality in real world applications is still limited to a small number of cases.

This work focuses on projector-based augmented reality for user feedback and an infrared tracking system for information input.

## 2.2 3D Graphical interfaces for industrial robotics

The use of 3D graphical interfaces in industrial robotics is very common in pre-programming operations (like p.e. [15]) that are mainly found in the office. This work targets a different use, making use of 3D graphical interfaces to improve the quality of the interaction with the workcell in the tuning and adjustment stages. In this scenario features such as collision avoidance or advanced path planning are less important when compared with the intuitiveness and ease of use, due to the different type of end-user.

## 3. Workcell concept

The workcell final layout is an aluminium-based, lightweight, framed structure mounted on a modular and portable steel structure base allowing fast setup and easy shop-floor layout re-engineering changes. All non-accessible areas are fully covered by transparent acrylic glass panels for clear viewing, ceramic dust protection and easy maintenance (Figures 3, 4 and 5).

The development of this workcell for the asymmetrical fettling workcell took into consideration up-to-date standards concerning ergonomics and safety. The position of the touch-screen monitor that shows the graphical user interface for system operation was carefully chosen due to the relation between the view-point and the perspective shown in 3D, which is explored in the graphical application presented in this section.

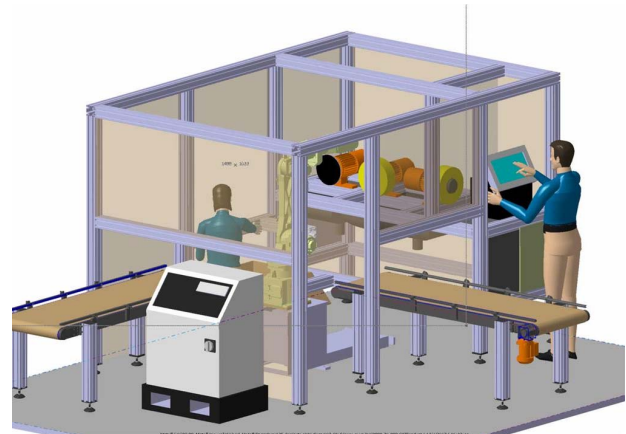


Figure 3. Final workcell.

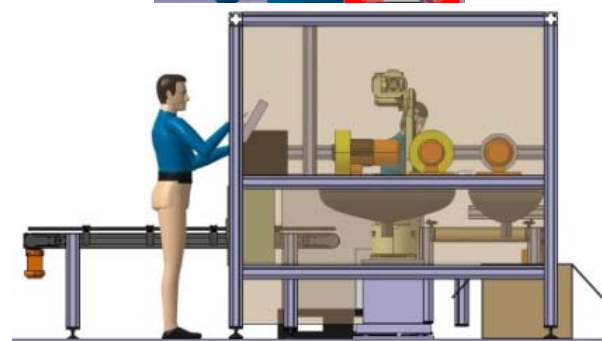


Figure 4. Operator's point of view.

The workcell includes a FANUC M16iB mechanical unit connected to a System R3iC [16] controller (a last generation controller available from FANUC). The mechanical unit is positioned centrally and on the front side of the cell three tooling devices are mounted facing the robot. On both sides there are input/output entrances from which parts are loaded and unloaded.



Figure 5. Prototype workcell.

All three tools are similar in the motion concerned since all are electric rotational tools based on 1Hp AC motors. The first tool is a grinding tool equipped with a radial grinding disc. The other two are fettling units equipped with soft but thick round sponges. Both feature a recirculatory water circuit for sponge wetting. All three rotational tools feature speed control and start and stop rotation instructions as well as speed set-points, which are set through the robot controller using analogue IO.

A vacuum gripper was used to hold the part. This gripper consists of a single "bellows" special suction type powered by a vacuum generator with a pulse ejector and vacuum switch. The vacuum generator is also controlled by the robot controller using digital IO.

This process was implemented in five stages: part pick-up, grind, fettle vertically, fettle horizontally and unload.

The isostatically produced ceramic parts are dropped on to a conveyor device, which transports and roughly aligns the part so that a transfer device can then be implemented to deliver the part in the centring, mastering bed. The robot then picks-up the part, now correctly aligned and positioned at known coordinates. The robot first forces the part through an "L" shaped mask, which instantly removes the excess material rim created by the pressing process, completing stage 1.

In stage 2, the robot moves the part on to the flexible motorized grinding tool, ensuring that the ceramic part edge maintains contact with the grinding disk during the entire planned trajectory and gets its peripheral roughness eliminated. In phase 3 and 4, the robot takes the part either to the vertical or the horizontal fettling unit, or both depending on the part at hand. During these stages, the robot moves the part allowing its contour to stay sunk down on the sponge, building a smooth finish and rounding the part edges (Figure 6).



Figure 6. Grinding and fettling stages.

The process ends at stage 4, in which the robot places the finished part on an output tray, returning to the initial home position, ready to process the next part (Figure 7).



Figure 7. Part Loading and unloading.

#### 4. Developed software

The higher requirements for this application were related to the human-robot interaction during the definition of new shapes (Section 3.1) and the daily operation of the system that includes workcell control and adjustment (Section 3.2).

##### 4.1 Augmented reality part programming system

One of the requirements from the end-user was the capability to define robot paths that were not directly related with the part geometry, as described in Section 1.2. Therefore, a data structure was defined to define paths with three fields: X, Y and the type of point. This data structure can be saved in two file formats: .xml (*Extendable Markup Language*) and .csv (*Comma Separated Values*). Following another end-user requirement the .xml format allows the user to compose or edit the data using Microsoft Excel and validate it against a premade schema. The *type of point* field defines the type of movement intended for the robot at that point. In Figure 8 an example is shown of a grinding operation with a disc that is not suitable for continuous robot movement in order to maintain live vertices on the plate. In this case the user defines four different trajectories with approach and escape points in positions placed outside the plate contour. The software application uses a specific algorithm to add the necessary robot fly-points to the path in order to keep the robot distance to the tool during the transition fly points. This algorithm also translates contour positions into robot positions and calculates the relative speed of each point of the trajectory in order to keep the relative speed of the plate over the tool constant.

To define the robot paths for a specific part a programming table was developed.

This table includes an infrared machine vision system that locates a led present in a pen and a projector that augments the reality with a software application that helps the user defining tool trajectories over the plate (Figure 9).

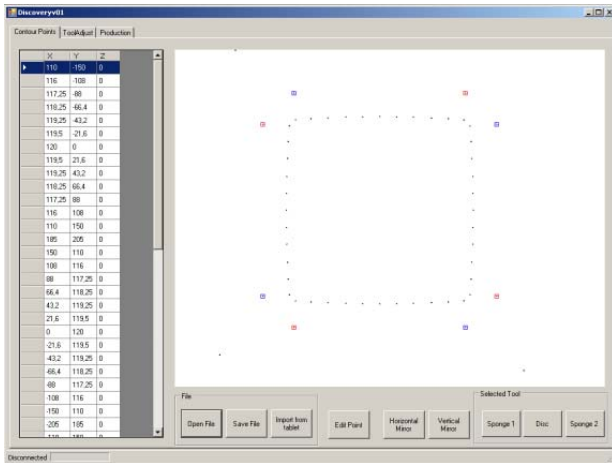


Figure 8. Part Loading and unloading.

The calibration of the system uses planar homographies as described in [17]. This calibration assumes that the projection screen is flat and that the optics from the projector and the camera can be modelled by perspective transforms, and allows the calibration without any previous knowledge of the positions or orientations of the projector and the camera. Considering a point  $(x, y)$  in the projector frame and a point in pixels  $(X, Y)$  in the camera frame there is a single projective transform that maps both items:

$$\begin{bmatrix} xw \\ yw \\ w \end{bmatrix} = \begin{bmatrix} p1 & p2 & p3 \\ p4 & p5 & p6 \\ p7 & p8 & p9 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

The matrix  $p$  can be determined from as few as four correspondences, calculating the eigenvector that corresponds to the smallest eigenvalue of  $A^T A$  where  $A$  is the following matrix:

$$\begin{pmatrix} X_1 & Y_1 & 1 & 0 & 0 & 0 & -X_1x_1 & -Y_1y_1 & -x_1 \\ 0 & 0 & 0 & X_1 & Y_1 & 1 & -X_1x_1 & -Y_1y_1 & -y_1 \\ X_2 & Y_2 & 1 & 0 & 0 & 0 & -X_2x_2 & -Y_2y_2 & -x_2 \\ 0 & 0 & 0 & X_2 & Y_2 & 1 & -X_2x_2 & -Y_2y_2 & -y_2 \\ & & & & & \vdots & & & \\ X_n & Y_n & 1 & 0 & 0 & 0 & -X_nx_n & -Y_ny_n & -x_n \\ 0 & 0 & 0 & X_n & Y_n & 1 & -X_nx_n & -Y_ny_n & -y_n \end{pmatrix}$$

Using the infrared vision system described before, it is very convenient to obtain the correspondent points by making a manual calibration process where the user/installer is guided through four or more calibration points.

The accuracy obtained in our system, which has an indicative angle of  $20^\circ$  between the projector and the vision system, which uses a  $1024 \times 768$ , is below 1mm for a projection surface with a diagonal of 600mm.



Figure 9. Programming table for product direct programming with infrared pen and augmented reality.

#### 4.2 3D graphical interface

One of the goals of this work was the development of an intuitive interface for the online tuning of the ceramic production system. This application permits the adjustment of the global position of the robot coordinate systems associated with a specific tool with the help of 3D graphics. Following the concept presented in Section 2.2, this 3D interface is used in a position such that the user sees the 3D representation of the system in a position equal to the real world (Figure 10).

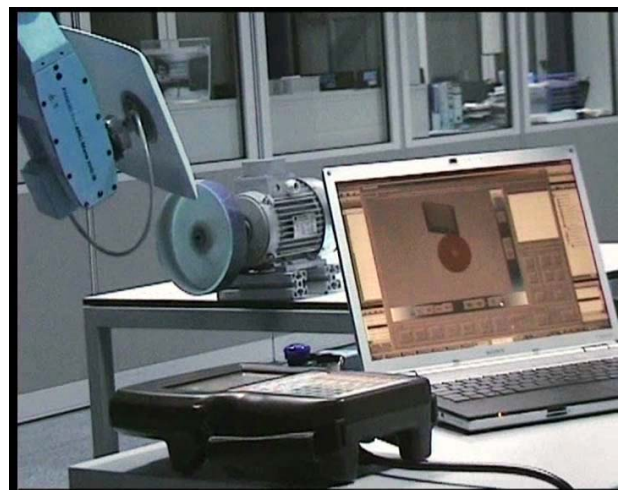


Figure 10. Online adjustment system.

When the user starts interacting with this interface the 3D View is the normal view (Figure 11), which is very close to the physical view of the system by the user.

This interface provides the user with different views of the tool/plate. To adjust different fettling angles the user interacts with only four arrow keys, the meaning of which is dependent on the selected view.

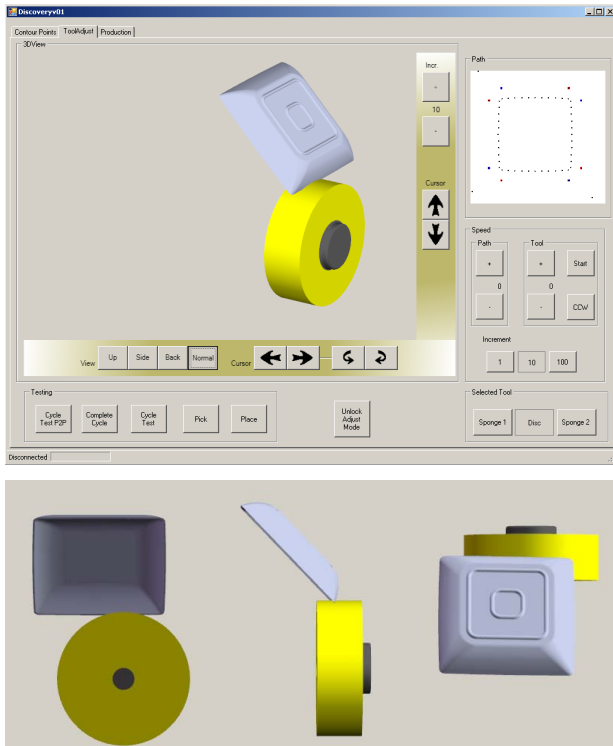


Figure 11. 3D graphical interface with the 3D isometric normal view (top) and different views for tuning (bottom).

This application also permits the management of all kinds of workcell procedures like start/stop motors, adjust motor speed, make robot pick/drop a piece and execute cycle tests. Another tuning procedure that deserves a closer look is available via the dialog box presented in Figure 12.

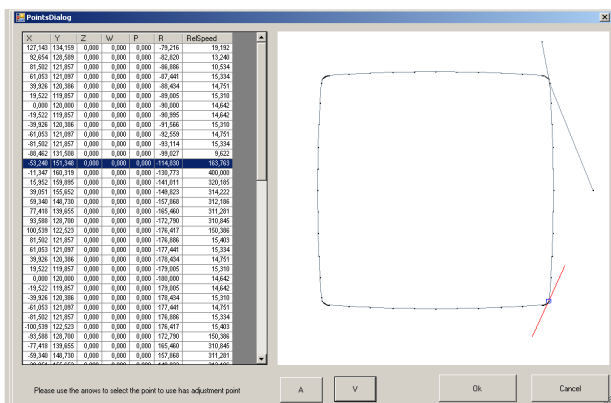


Figure 12. Trajectories correction screen.

This application was made using the graphical framework *Windows Presentation Foundation* (WPF) because it supports extensive 2D interaction elements but also 3D elements.

This software application was developed with service-oriented principles applied to the device level using UPnP [18]. Services were orchestrated using a

SidneyChart [19], which permitted the necessary separation of concerns between the execution and the coordination [20].

## 6. Evaluation tests

To evaluate the performance of the proposed robot programming system a study was conducted with five operators from the ceramic industry that did not have previous contact with the current work/cell, although they had different experience in terms of robotics. Two second year Engineering students without previous formal contact with robotics were also used. Each operator had to program five tableware parts, which involved programming three grinding operations, three fettling operations and realizing all the necessary adjustments both in terms of adjusting misplaced points and also in terms of the overall tool coordinate system (User Frame in the FANUC's terminology). These operators performed these tasks randomly using the offline programming tool FANUC RoboGuide 6.40b [21] the robot console and the newly proposed programming system. In the offline RoboGuide tool the operator was taught to use the CAD-to-path feature that automatically generates robot trajectories from CAD files.

Figure 13 shows that the presented system outperforms the other systems in terms of time required for the system to learn a new plate and to tune the distance to the tools. Furthermore, the standard deviation indicates that our system is less dependent on natural or previously acquired competences by the operators.

The required time to program the system with the tool presented in this work also outperforms offline and teach pendant programming.

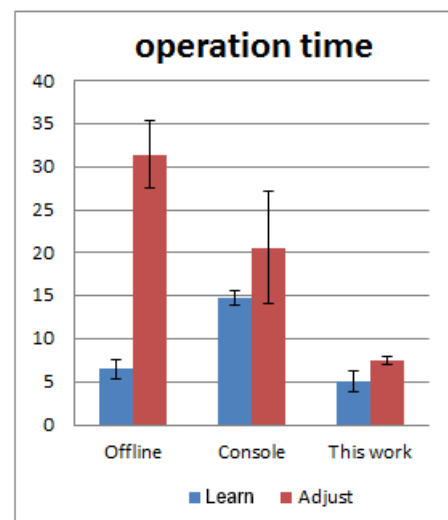


Figure 13. Programming times comparison (hours).

A second evaluation measure is the time required to learn how to use the tool. Table 1 shows the time required to learn how to program a new product and operate the system safely. It should be mentioned that the time required both for the console and the offline programming system were the absolute minimum required to program the parts, not to become a proficient with the tool.

	Learning time (hours)
Offline	192
Console	96
This work	3

**Table 1.** Table presenting operator learning time for each programming system.

## 7. Conclusions

The robotic workcell for ceramic part finishing presented in this work meets demanding requirements in terms of product quality, process flexibility, finishing precision, machine availability and efficiency of the ceramic industry. The development of this application was made in close cooperation with the end-user due to the high requirements of ease-of-use. A video of the system running can be seen at <http://www.sarkkis.com/mechatronics/products/keramica>.

The main original features of this application are the highly flexible programming model and the overall intuitiveness of the system operation.

The results show that this application outperforms the traditional programming techniques, namely offline programming and teach pendant programming, not only in terms of the time required to program and tune the system but also in the time required to learn how to operate the tool.

Future work will include a complete long-term evaluation with a large number of end-users.

## 8. References

- [1] Zengxi, Pan and Zhang, Hui, "Robotic machining from programming to process control: a complete solution by force control," *Industrial Robot: An International Journal, Emerald*, vol. 35, no. 5, 2008.
- [2] M. Ferreira, L. Rocha, P. Costa, and A. Moreira, "Stereoscopic vision system for human gesture tracking and robot programming by demonstration," in *Proceedings 2013 Flexible Automation & Intelligent Manufacturing*.
- [3] R. D. Schraft and C. Meyer, "The need for an intuitive teaching method for small and medium enterprises," presented at the ISR 2006, 2006.
- [4] Pires, J. N., Veiga, G., and Araujo, R., "Programming by demonstration in the coworker scenario for SMEs," *Industrial Robot, Emerald*, vol. 36, no. 1, 2008.
- [5] H. Zhang, H. Chen, N. Xi, G. Zhang, and J. He, "On-Line Path Generation for Robotic Deburring of Cast Aluminum Wheels," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, 2006, pp. 2400–2405.
- [6] B. Solvang, G. Sziebig, and P. Korondi, "Robot Programming in Machining Operations," in *Robot Manipulators*, M. Ceccarelli, Ed. InTech, 2008.
- [7] Z. Pan, J. Polden, N. Larkin, S. V. Duin, and J. Norrish, "Recent Progress on Programming Methods for Industrial Robots," in *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*, 2010, pp. 1–8.
- [8] J. N. Pires and S. Paulo, "High efficient robotic de-palletizing system for the non-flat ceramic industry," presented at the Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on, 2003, vol. 3, pp. 3529–3534 vol.3.
- [9] O. Cakmakci and J. Rolland, "Head-worn displays: a review," *Display Technology, Journal of*, vol. 2, no. 3, pp. 199–216, Sept.
- [10] O. Bimber and R. Raskar, *Spatial Augmented Reality: Merging Real and Virtual Worlds*. Natick, MA, USA: A. K. Peters, Ltd., 2005.
- [11] H. Regenbrecht, G. Baratoff, and W. Wilke, "Augmented reality projects in the automotive and aerospace industries," *Computer Graphics and Applications, IEEE*, vol. 25, no. 6, pp. 48–56, 2012.
- [12] J. Zhou, Lee, Ivan, B. Thomas, R. Menassa, A. Farrant, and A. Sansome, "In-Situ Support for Automotive Manufacturing Using Spatial Augmented Reality.," *International Journal of Virtual Reality*, pp. 33–41, 2012.
- [13] J. W. S. Chong, S. K. Ong, A. Y. C. Nee, and K. Youcef-Youmi, "Robot programming using augmented reality: An interactive method for planning collision-free paths," *Robot. Comput.-Integr. Manuf.*, vol. 25, no. 3, pp. 689–701, Jun. 2009.
- [14] R. Marin, P. J. Sanz, and J. S. Sánchez, "A very high level interface to teleoperate a robot via Web including augmented reality," in *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, vol. 3, pp. 2725–2730.
- [15] "Visual Components - Configuration, Visualisation, Simulation, Optimization." [Online]. Available: <http://www.visualcomponents.com/index.php?id=141>. [Accessed: 14-Jan-2009].
- [16] "FANUC Robotics SYSTEM R-J3iB Controller KAREL Reference Manual," Reference Manual, 2002.
- [17] R. Sukthankar, R. G. Stockton, and M. D. Mullin, "Smarter Presentations: Exploiting Homography in Camera-Projector Systems," in *ICCV'01*, 2001, pp. 247–253.

- [18] M. P. Bodlaender, "UPnP™ 1.1 - designing for performance & compatibility," *Consumer Electronics, IEEE Transactions on*, vol. 51, no. 1, pp. 69–75, 2005.
- [19] P. Malaca, G. Veiga, and N. Pires, "SidneyChart: A statechart GUI for SOA orchestration in autonomous industrial systems," in *Industrial Informatics (INDIN), 2011 9th IEEE International Conference on*, 2011, pp. 238–243.
- [20] G. Veiga, P. Malaca, K. Nilsson, and J. N. Pires, "Separation of concerns on the orchestration of operations in flexible manufacturing," *Assembly Automation*, vol. 32, no. 1, 2012.
- [21] R. FANUC, "FANUC Robotics Roboguide V6.40 Rev.B Accompanying Training Manual,"