The International Journal of Advanced Manufacturing Technology
https://doi.org/10.1007/s00170-018-03282-w

ORIGINAL ARTICLE

Automatic robot path integration using three-dimensional vision and offline programming

Amit Kumar Bedaka[1] · Joel Vidal[1] · Chyi-Yeu Lin[1,2,3]

Received: 20 June 2018 / Accepted: 28 December 2018

Abstract

In manufacturing industries, offline programming (OLP) platforms provide an independent methodology for robot integration using 3D model simulation away from the actual robot cell and production process, reducing integration time and costs. However, traditional OLP platforms still require prior knowledge of the workpiece position in a predefined environment, which requires complex human operations and specific-purpose designs, highly reducing the autonomy of the systems. The presented approach proposes to overcome these problems by defining a novel automated offline programming system (AOLP), which integrates a flexible and intuitive OLP platform with a state-of-the-art autonomous object pose estimation method, to achieve an environment and model independent platform for automatic robotic manufacturing. The autonomous recognition capabilities of the three-dimensional vision system provide the relative position of the workpiece model in the OLP platform, with robustness against clutter, illumination, and object material. After that, the user-friendly OLP platform allows an efficient and automatic path generation, simulation, robot code generation, and robot execution. The proposed system precision and robustness are analyzed and validated in a real-world environment on four different sets of experiment. Finally, the proposed system's features are discussed and compared with other available solutions for practical industrial manufacturing, showing the advantages of the proposed approach. Overall, despite sensor resolution limitations, the proposed system shows a remarkable precision and promising direction towards highly efficient and productive manufacturing solutions.

Keywords Automated offline programming · Path generation · Industrial manipulator · Machine vision · 3D object recognition · 6D pose estimation

✉ Chyi-Yeu Lin
jerrylin@mail.ntust.edu.tw

Amit Kumar Bedaka
akbedaka@gmail.com

Joel Vidal
jolvid@gmail.com

1 Department of Mechanical Engineering, National Taiwan University of Science and Technology, Taiwan, Republic of China

2 Taiwan Building Technology Center, National Taiwan University of Science and Technology, Taiwan, Republic of China

3 Center for Cyber-Physical System Innovation, National Taiwan University of Science and Technology, Taiwan, Republic of China

1 Introduction

Industrial manufacturing has long relied on human operators to perform challenging and skilled tasks. Certainly, the introduction of industrial robotic systems has dramatically improved the production level, taking over most tasks with a predefined and repetitive nature. Applications such as pick and place, welding, painting, or gluing are some of the common jobs carried out nowadays by robots. However, these tasks are usually programmed using conventional methods, like using a teach pendant, in an online programming manner on highly constrained environments. These integration processes require stopping the workcell while expert operators program in situ the robot actions and trajectories for each specific task, spending a big amount of time, and resources on nonflexible solutions with a very limited range of applications.

In view of the time, costs, and difficulties associated with the manual online programming and reprogramming of industrial robotic systems in manufacturing scenarios, methods based on CAD model simulation of robot integration arise, known as offline programming (OLP). These software platforms provide all the necessary tools for a complete and realistic simulation of the robot manufacturing environment by using precise CAD model designs. Using these simulation platforms, the robot program can be carefully designed, planned, tested, and generated out of the workcell, only requiring a brief stop for the final program download. In [41], Mitsi et al. present an OLP system, including graphical simulation, robot kinematics, motion planning, and automatic code generation for welding operations. Additionally, Larkin et al. [34] evaluate several OLP software packages used for welding, including ABB Robotics, Delmia from Dassault Systems, a Matlab-based OLP system, and RinasWeld from Kranendonk Production System.

In the last decades, a significant amount of research has been done using commercial OLP platforms, such as KUKA Sim for Kuka, RobotStudio for ABB, MotoSim for Motoman, Delmia from Dassault Systems, RobCAD from Technomatix Technologies, and Robotmaster from Jabez Technologies [3, 48]. As an example, the OLP system proposed by [59] joins the geometric functions of CATIA (e.g., curve/surface intersection, a projection of the points onto the surface, etc.) with the simulation function of KUKA Sim Pro (e.g., robot kinematics, collision detection, etc.). Their method focused on robotic drilling applications in aerospace manufacturing, improving the position accuracy by using bilinear interpolations model and redundancy resolution. Despite their efficiency, most commercial solutions are subjected to high-cost licenses and a limited range of applications and improvement. Some alternative solutions propose commercial general-purpose CAD packages to define more flexible and cost-effective platforms. Neto et al. explored the most suitable way to represent the robot motion in CAD drawings using Autodesk Inventor, the automatic extraction of the motion data and the mapping between virtual and real environment to generate the robot program [42, 43]. Similarly, some platforms proposed to integrate mechanical CAD features and robotics CAD models with SolidWorks Application Programming Interface (API) [7, 37]. Regardless of their advantages with respect to online programming methods, the OLP platforms highly depend on a precisely defined workcell and still requires a significant amount of human operation, including the selection of tag points (i.e., start and end positions) for path planning and to solve singularity- and collision-related problems.

Recently, Automated Offline Programming (AOLP) systems are gaining attention in research, providing autonomous alternatives to manual or semi-automatic OLP tasks [45]. These platforms provide significant advantages such as automated modeling of the environment or singularity-free trajectories by means of additional sensors and advanced techniques. Ames et al. [5] develop an AOLP solution to automatically generate complete robot programs without programming requirements to perform welding tasks. Similarly, Polden et al. [46] presents an automatic module for Delmia that provides automatic tag generation and trajectory planning stages for welding applications. Both systems can generate collision-free and singularity-free trajectories for the complete working path. However, the aforementioned AOLP systems still rely on precise CAD geometry of the working environment, defining error-prone and nonflexible solutions that require a high level of human intervention. These systems require a specifically designed environment or manual calibration of the CAD models for each different workpiece and environmental setup, which become not cost-effective in the long run. In this respect, the visual understanding of the environment is an important step towards a more flexible and efficient system.

In the manufacturing and production industries, machine vision has been widely used in different applications [40]. Many types of vision systems and sensors have been proposed to provide reliable solutions according to the particular requirements of individual applications. Some common examples include automatic inspection [44] or robot guidance [20]. In particular, the object recognition problem has been deeply studied towards fully autonomous systems which can work independently of human operators [6]. Solutions based on 2D visual object recognition have been successfully applied for simple pick and place operations or random bin-picking tasks [56]. However, these methods are still affected by environment illumination changes, background clutter, and low robustness. On the other hand, 3D recognition systems based on range data have been proposed, which are robust to illumination and show relatively good results under clutter and occlusion environments [25]. These methods are reviewed in detail, in Section 3.

Recently, these intelligent vision-based solutions have advanced on autonomous manufacturing techniques, which do not require robot programming. In this direction, open-source robotics frameworks such as ROS Industrial [1] provide an extensive set of packages for autonomous manufacturing and simulation. However, these autonomous methods face several yet-to-be-solved intrinsic challenges related to real-time decision and sensors data interpretation, such as working range or camera view limitations, which are not faced by OLP systems. These challenges generate severe difficulties to the real applicability of these autonomous solutions [11], which still limit their application to simple tasks, like pick and place [18, 22], or constrained environments [30]. In this sense, AOLP systems still stand out as a compromise solution for the manufacturing industry.

Vision-based techniques have also been proposed to automate OLP functions. Larking et al. proposed to use Time of Flight (TOF) sensors to map the workcell 3D environment for motion planning without using CAD data [35], effectively avoiding specific-purpose designs and CAD models calibration. However, their system does not provide precise information about the workpiece for automated manufacturing planning operations. In a different direction, Maiolino et al. proposed an AOLP solution for workpiece detection [39]. Their method connects the functions of a commercial offline RobotStudio software, by means of a specifically developed add-on, with an RGB-D recognition module using a UDP socket interface. Their work only analyzed the sensor performance in different illuminations and did not provide results regarding the performance of the system. One of their method's limitations is the nature of the object recognition approach, which requires an isolated object and relies on a segmentation step. This preprocessing step, based on plane filtering, increases the system complexity and may limit its performance. Another crucial limitation comes from the proposed system architecture using a UDP socket communication between the vision module and the commercial software. This add-on solution limits the system applicability and extension capability, making the platform unsuitable for advanced fully integrated tasks and control strategies that are required for complex intelligent robotic manufacturing, such as visual servoing [36] or automatic inspection [10] techniques.

This paper presents a novel and more flexible AOLP solution that fully integrates a state-of-the-art 6D pose estimation approach into a flexible OLP platform, defining a novel solution that does not require the manual calibration of the workpiece position and can be used or integrated with other systems for advance intelligent manufacturing implementations. In detail, the contribution of this work is a novel AOLP system in an integrated modular architecture, which joints the benefits of latest three-dimensional vision recognition with a versatile OLP platform, proposing a more efficient and flexible solution to overcome the aforementioned limitations. In contrast with other OLP and AOLP approaches, the proposed solution does not require stopping the workcell, complex workpiece calibration procedures and specifically designed or constrained environments to determine the target workpiece in the robot cell. In the one hand, the autonomous three-dimensional vision module determines the position of the object in unconstrained scenes, in a global manner, without requiring pre-segmentation steps. On the other hand, the proposed fully integrated AOLP architecture overcomes the limitations of previously proposed approaches [38, 42] by allowing our system to be coupled with other advanced intelligent solutions. Some potential applications include high precision tasks using visual servoing [36] and integrated manufacturing process with automatic optical inspection (AOI) [10], which require a fully integrated framework. To archive this characteristics, the proposed system has been based on a noncommercial and cost-effective OLP solution developed on Open Cascade open-source libraries, including an efficient path generation with automated tag creation from CAD primitives. On top of that, the AOLP platform bases its three-dimensional vision capabilities on a highly reliable Point Pair Features (PPF) approach [57] that allows an efficient and robust autonomous localization of the workpiece. The proposed system effectiveness and robustness is evaluated in a real-world environment with two different methodologies. First, the relative error of the system is computed for the X-, Y-, and Z-directions. Second, the absolute error of the system is evaluated for multiple random poses against a human-defined ground truth. The method robustness is discussed and analyzed with additional experiments for different illumination and object materials. Finally, the overall system features and precision are compared with other existing methods, which reveals the advantages of the proposed system with respect to the other available solutions. Overall, the presented system defines a novel, flexible, and efficient fully integrated platform that focuses on reducing integration time and increasing productivity in manufacturing.

The rest of the paper is organized as follows: Section 2 provides an overview of the system architecture, the range sensor, and OLP system. Section 3 reviews the object recognition state-of-the-art and introduces the proposed algorithm. Section 4 explains the OLP system development and the integration with the vision system. Finally, Section 5 presents the results of real-world experiments and Section 6 closes with the conclusions.

2 System overview

Offline programming (OLP) systems are semi-automated platforms that rely on accurate CAD designs to simulate industrial manipulator manufacturing tasks out of the workcell, in order to avoid costly and time-consuming procedures on the robot production line. This CAD information is usually provided through specific-purpose designs or complex calibrations of the robot environment. In the presented novel automated approach, a depth sensor (Kinect) is employed to extract three-dimensional information of the environment to autonomously locate the workpiece on the workcell, regardless of illumination and background clutter. This object recognition capability is integrated with a user-friendly OCC-based OLP platform, which includes an automatic path planning based on CAD information, in order to plan, simulate, analyze, and generate robot control code for manufacturing process on an industrial manipulator (Denso 6556). The architecture of the proposed system is presented in Fig. 1.

2.1 Kinect sensor

Kinect is an RGB-D sensor, capable of providing registered color and depth data, introduced in 2010 by Microsoft as a game device for the Xbox 360 platform. In 2012, a similar version, named "Kinect for Windows," was released for commercial use.

The sensor is based on infrared (IR) structured-light technology [24]. As shown in Fig. 2, the system uses one IR laser projector and two cameras. One camera is used to capture the color image (RGB data) and the other camera, designed to capture only IR light, is used to extract the depth data from the projected IR structured pattern. In addition, the sensor has a microphone array and a tilting motor.

Due to its relatively good precision, considerably high frame rate and low-cost, the sensor quickly became popular in research fields beyond computer entertainment. In addition, the usage of IR light for active sensing, provides depth data independent of visible illumination, working even in absence of light. These characteristics make the Kinect sensor a good choice for the proposed system. Nevertheless, any other range sensor with similar or better characteristics can be employed.

2.2 Offline programming platform

The proposed OLP platform [8] is based on the OCC library, which provides open-source powerful CAD kernels. The library has become a standard solution for the design and development of open-source application oriented to CAD design and OLP platforms. The proposed system is built by the integration of a user-friendly GUI and OCC-based OLP tools. Using simple mouse interactions, the robot trajectories are automatically generated by extracting and processing the CAD features of a workpiece with advanced techniques embedded in the OCC libraries, without the need of any commercial CAD packages. In addition, the proposed platform uses a virtual environment and simulates the robot trajectory in order to check issues related to the manipulator's reachability, possible collision along the path and singularities. After simulation, the robot program can be generated and sent directly to an industrial robot manipulator.

3 Recognition and pose estimation

Imaging-based inspection and analysis techniques, commonly known as machine vision (MV), have become an essential part of industrial automation with a wide range of practical applications [40], including tasks of automated inspection [44] or robot guidance [20]. In this context, recognition and pose estimation of rigid three-dimensional objects is a key part to accomplish highly complex autonomous tasks, commonly involving object grasping or manipulation in flexible environments.

Recognition and detection techniques on 2D images have been successfully developed for relatively flat objects [56] and textured objects [23, 55]. Despite most texture-based 2D techniques having a remarkably high level of recognition, their capabilities are limited to detection and 2D localization, which is insufficient for manufacturing purposes. In addition, industrial environments are commonly

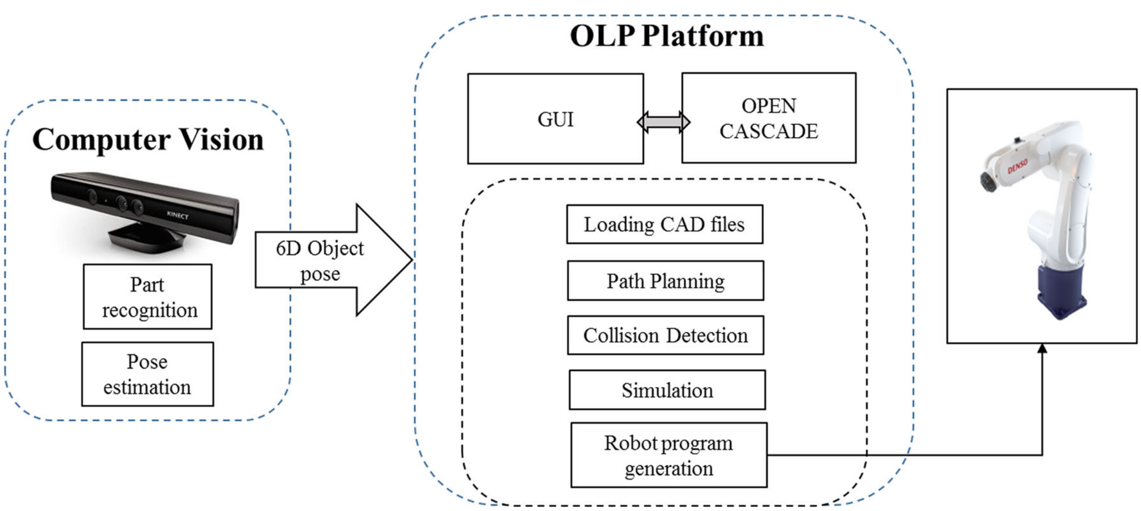


Fig. 1 Architecture of the proposed platform

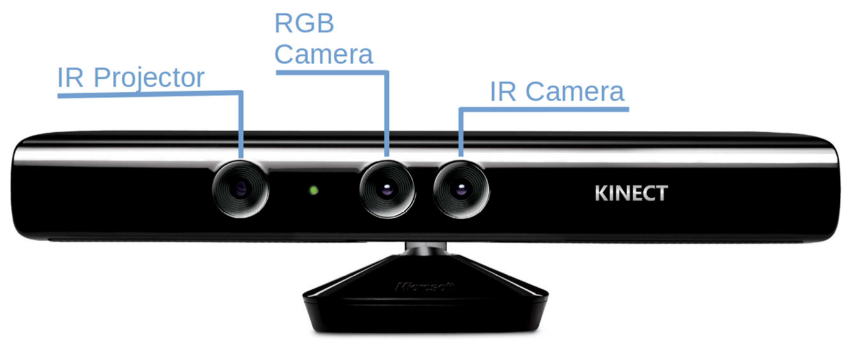


Fig. 2 Kinect's hardware

populated with three-dimensional textureless objects, which still remains a challenging case. Only recently, novel machine learning approaches [32] have shown ability to solve the 6D pose estimation problem for textureless objects, obtaining promising results. As an alternative to the traditional monocular solution, range sensors and 3D computer vision algorithms have been proposed to solve the object recognition problem, providing 6D pose estimation methods more robust to complex geometries and textureless cases. The current state-of-the-art for 6D object pose estimation methods using RGB-D data can be classified into three categories: feature-based, template matching, and machine learning methods. Feature-based methods are usually divided into local and global approaches. Local approaches [25] are based on local descriptors encoding the surface characteristics of the object around selected key points. On the other hand, Global approaches [4, 51, 58] encode the object surface information as a whole, usually in a view-dependent manner. Template matching solutions like [26, 27] use multimodel templates captured from model objects by joining image gradients and surface normals. Finally, machine learning methods using Convolutional Neural Network (CNN) [31] and random forest [14] have been proposed. Overall, these three-dimensional techniques have shown better results than the 2D approaches regarding the 6D pose estimation problem.

As a special case in the feature-based category, the point pair features (PPF) voting approach, first proposed in 2010 [21], combines a global modeling and local matching using spears descriptors. The method relies on these features created between a pair of oriented points to globally describe the whole object model surface. This four-dimensional feature, represented in Fig. 3, is defined by Eq. 1,

$$F(p_1, p_2, n_1, n_2) = [||d||, \angle(n_1, d), \angle(n_2, d), \angle(n_1, n_2)]^T, \quad (1)$$

where p_1 and p_2 are a pair of 3D points, n_1 and n_2 are their respective normals, $d = p_2 - p_1$, and $\angle(a, b)$ defines the angle between vectors a and b. The method is divided into two stages: modeling and matching. During modeling, the method defines a four-dimensional lookup model table by quantizing the model feature space. During matching, this model table is used to match quantized features from scene pairs. For each scene pair, the obtained scene-model pair correspondences are grouped in a two-dimensional voting table, where the table peak represents the most likely pose, defining a pose of the object model to the scene. Finally, all poses obtained from different scene points are clustered to define a final hypothesis.

Recently, Vidal et al. [57] proposed an improved method based on the PPF voting approach, combining more efficient preprocessing techniques with improved matching methodologies and robust verification steps. Their method was tested on multiple challenging scenarios with 68 objects and more than 60000 test images, including industrial-like objects and random bin pinking cases. The method showed, for the same set of parameters, a high level of robustness on multiple objects and uncontrolled scenarios under clutter. In particular, datasets with minimum occlusion, an expected condition for industrial setups, shows recognition rates up to 100% recall for different types of object and thousands of test images. For occluded cases, unlikely for industrial environments, show a remarkable recognition rate near 70% recall for industrial-like objects. These results demonstrate the good performance of the method under background clutter scenarios. The method's robustness has been recently tested against other 14 state-of-the-art solutions on the extensive BOP benchmark [28], showing to outperform all methods with the best overall performance. In addition, compared to other evaluated approaches [14, 27, 31], the method does not use color information, relying on depth data only. Therefore, the method works independently of the light conditions when used alongside an IR-based structured-light sensor, like Kinect.

For this system, we propose to use a variation of the method presented in [57], including an additional point-to-plane iterative closest point (ICP) [12, 17] refinement step to ensure the maximum accuracy from the sensor data. In detail, the optimal transformation T_{fit} is defined by Eq. 2,

$$T_{\text{fit}} = \underset{T}{\text{argmin}} \sum_i ((Tm_i - s_i) \cdot n_i) \quad (2)$$

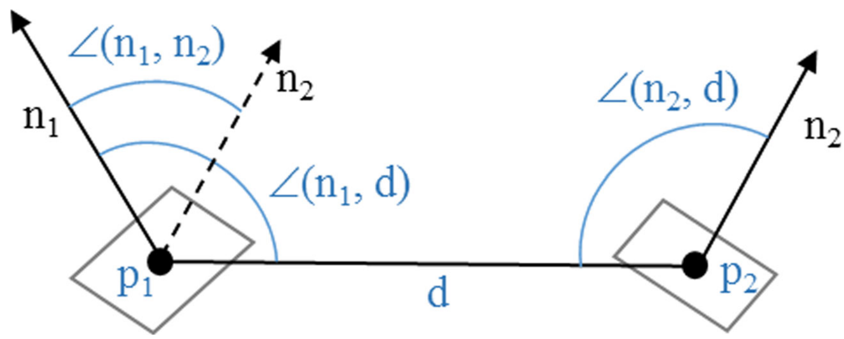


Fig. 3 Definition of point pair feature

where T is the model-to-scene transformation matrix, m_i is a point on the model surface, s_i is the scene destination point, and n_i is the unit normal on s_i. For each new iteration, the destination point s_i is defined as the nearest scene point to the last iteration transformed model point m_i.

Overall, this solution defines an accurate and reliable 6D pose estimation module, focused on the requirements of industrial scenes, robust to light and background changes with relative robustness to unexpected partially occluded situations. Figure 4 shows a high-level flowchart of the proposed solution, where the *modeling* and *matching* parts are based in [57].

4 AOLP path planning and generation

The AOLP automatic path planning system has been developed by the integration of the object recognition vision module with the flexible OLP platform. The flow chart of the proposed AOLP platform is shown in Fig. 5.

Initially, the OLP core system, which contains the CAD model and environment data of the process, requests the object pose to the vision module, which extracts the position and rotation of the object using the range sensor. This information is processed by the core system, which treats the pose with respect to the vision sensor frame. Consequently, the platform transforms the 6D pose data with respect to the robot frame and loads the relative object CAD model at the same position in the virtual environment. This task is performed autonomously, independently of the object pose and light conditions of the environment. After the recognition of the object pose, the OLP platform is able to generate the targeted path automatically extracting the CAD information.

4.1 CAD model processing

The pose estimation algorithm obtains and transmits precise information of the workpiece pose with respect to range sensor frame. This object pose information is represented as the homogeneous matrix ${}^{C}T_{O} \in SE(3)$, which represents the transformation of the object model to the camera frame, obtained from the three-dimensional recognition module. Using this transformation, the OLP core system further transforms the object pose from the camera frame to the robot frame, which defines the scene

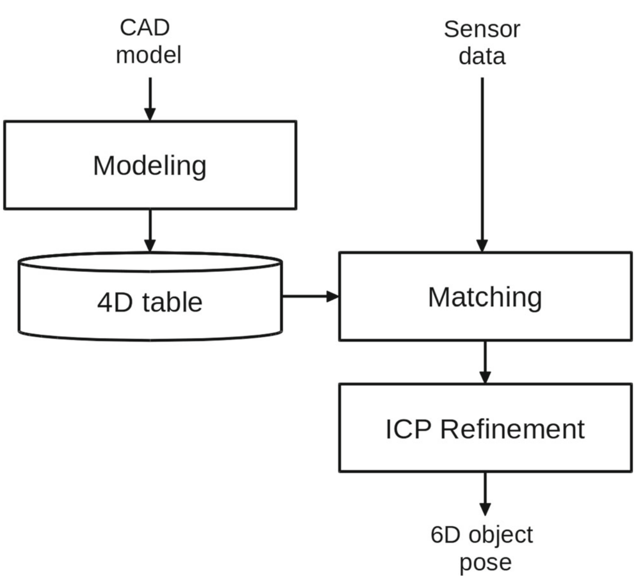


Fig. 4 Flowchart of the 6D pose estimation module

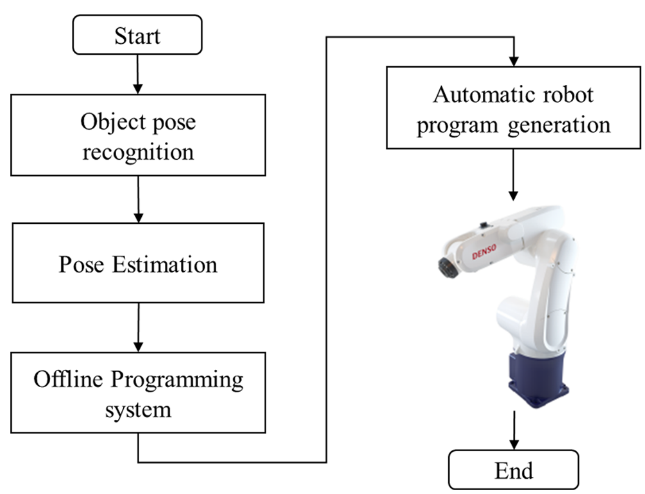


Fig. 5 Flowchart of the AOLP platform

global coordinate system, as shown in Fig. 6. This camera-to-robot transformation, ${}^{R}T_{C} \in SE(3)$, specifies the relative position and rotation of the range sensor with respect to the robot pose, which remains constant for any given industrial manipulator and range sensor fixed setup. This transformation is obtained by a calibration procedure for which a known pattern, i.e., a calibration grid, is attached to the robot end-effector and detected in several positions solving the well-known AX = XB equation [53, 54], analogously to the hand-eye calibration with a fixed camera and a moving calibration grid. In addition, methods using depth data can also be applied. The reader can refer to [19, 29, 53, 54] for a detailed explanation and solutions to the hand-eye calibration problem. Therefore, camera-to-robot transformation ${}^{R}T_{O} \in SE(3)$ is defined by Eq. 3.

$$ {}^{R}T_{O} = {}^{R}T_{C}\, {}^{C}T_{O} \tag{3} $$

After the object transformation is obtained, the OLP core loads the related CAD model to its actual position in the simulated environment, defining an accurate representation of the robot workcell.

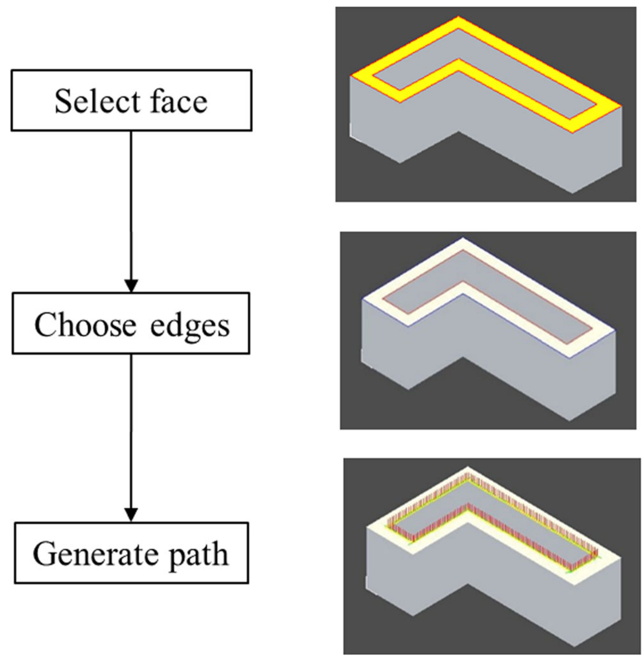


Fig. 7 Steps to generate a path

4.2 Path generation

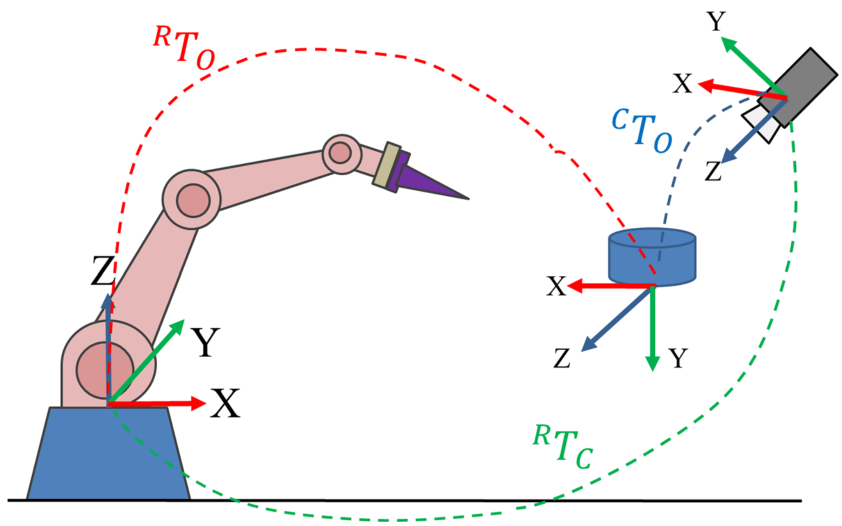
Once the CAD model (workpiece) is loaded into the environment, the robot path can be automatically extracted from the CAD information to perform the desired manipulation tasks. Based on the approach proposed by Amit et al. [8, 9], the user indicates the working tasks with respect to the CAD model structure using an intuitive and friendly interactive platform. Using a mouse or a tactile screen, the user can automatically extract path references (tags) by selecting desired abstract CAD features, such as a face, wire, edges, or vertexes, easily indicating the targeted working zone on the workpiece. For example, the user can select a face of the CAD object and indicate to work on its relative edges in few clicks, as shown in Fig. 7. Once, all target zones are selected, the related tag's edges are automatically analyzed and connected to generate a consistent path. For each path point, the orientation of the robot end-effector is defined by the roll, pitch, and yaw angles obtained from the orthogonal matrix defined with the path direction and underlying CAD surface. In detail, using α, β, and γ to represent the roll, pitch, and yaw angles, respectively, the angles are defined by Eqs. 4, 5, and 6,

$$ \alpha = \mathrm{atan2}(r_{32}, r_{33}) \tag{4} $$

$$ \beta = \mathrm{atan2}\left(-r_{31}, \sqrt{{r_{11}}^{2} + {r_{21}}^{2}}\right) \tag{5} $$

$$ \gamma = \mathrm{atan}(r_{21}, r_{11}) \tag{6} $$

Fig. 6 Camera and object position with respect to the robot



where r_{ab} represents the elements of the rotation matrix and $\mathrm{atan2}(y, x)$ a two-argument arctangent function that yields to one unique solution. Exceptionally, in the case of $\beta = \pm 90°$, a special singularity case given by the alignment of the x- and z-axis occurs, known as gimbal lock [47]. This situation is solved by applying Eq. 7,

$$\alpha \pm \gamma = \mathrm{atan2}(r_{23}, r_{13}) \tag{7}$$

where we use the default configuration of $\alpha = 0°$. Once the end-effector is defined, the angles for each joint are calculated using the inverse kinematics of the industrial manipulator.

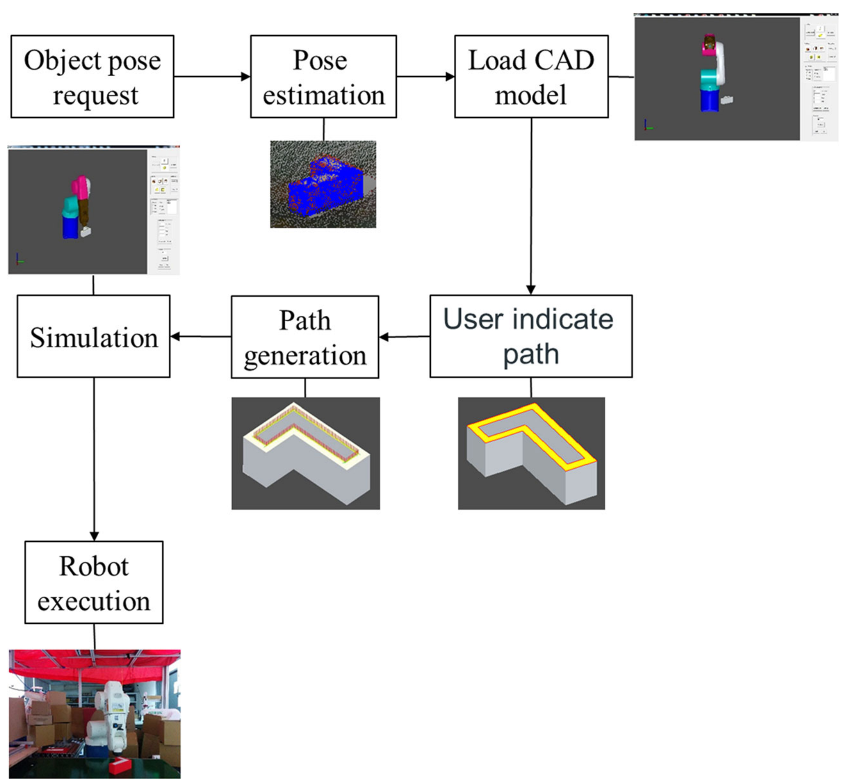
Using the robot inverse kinematics and working space constraints [33], the system is able to analyze the multiple positions and joint configurations along the path, sending information about the singularities and out of workspace issues. In these situations, the platform allows the user to redefine the working space and manipulation arrangement of the object to fit the robot constraints. Since the user can interact with the GUI interface, the proposed solution does not require the use of any programming language to generate a robot path simulation in the virtual environment. In more detail, during the robot path generation process, each of the target path steps is converted to robot angles taking special care of the singularities [13]. The path is automatically constructed with the desired discretization steps and rotation, following the requirements indicated in the GUI. After path generation, the virtual environment allows an accurate simulation and analysis of the robot performance without the use of the actual robot. Finally, the robot program can be generated to execute the planned trajectory on the real industrial manipulator.

Overall, the proposed AOLP platform allows the user to generate a robot path, simulation, and program mapped with an industrial manipulator with a few simple steps. The complete execution process is shown in Fig. 8. Initially, the OLP platform sends an acquisition request to the vision module in order to receive the object pose information with respect to the real environment. Consequently, the model is loaded in the same position within the virtual environment. At this point, the user makes use of the platform interface to indicate the desired working path. After these actions, the robot path is automatically generated. The simulation is performed to verify the correctness of all the robot movements before mapping it with the real industrial robot. Finally, the robot program is executed on the real robot performing the desired task.

5 Results

The proposed AOLP system was fully implemented in C++ using OCC [2] and Point Cloud Library (PCL) [50] open-source libraries. The system evaluation is divided into four sets of experiment. The first experiment evaluates the relative error of the system while moving the robot on X-,

Fig. 8 System's execution steps



Y-, and Z-direction. The second experiment evaluates the absolute error against a human defined ground truth. Third and fourth experiments analyze the robustness of the system for different illumination levels and object materials. All experiments have been conducted on a standard setup with the industrial robot manipulator working on a flat surface, as shown in Fig. 9. The Kinect sensor was placed at about 30–35° and 100 cm away from a Denso 6556 robot, within the best resolution distance and far enough to avoid obstructing the robot working space. Notice that all experiments have been conducted on scenarios with uncontrolled clutter. However, the robustness of the system against background clutter has not been analyzed in these tests as an exhaustive analysis and comparison of the object recognition module in terms of recognition rate, clutter, and occlusion performance can be found in [28, 57]. In addition, due to the nobility of the presented AOLP solution, relying on three-dimensional object pose estimation, to the best of our knowledge, no similar AOLP system experimentation results has been presented before. Therefore, we provide a comparison table with different method's features, discussing the strengths and limitations of the proposed method against other available solution for industrial manufacturing.

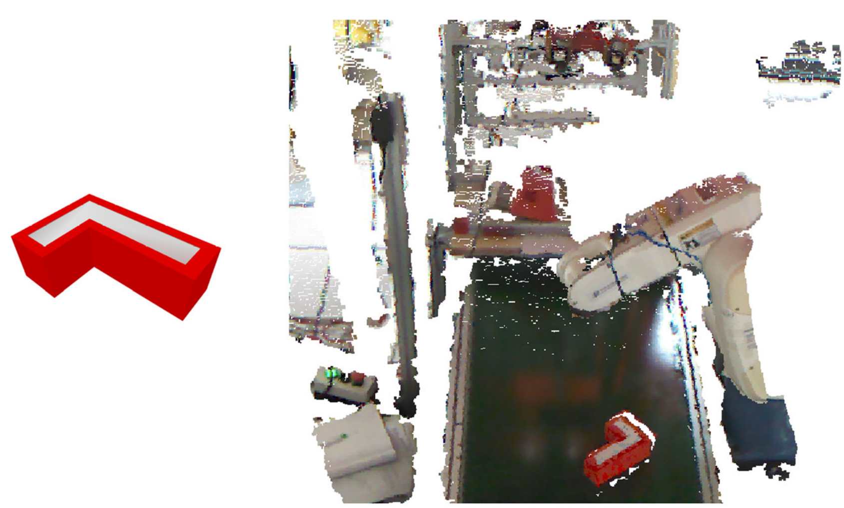
5.1 Evaluation of system relative error

First, the relative error of the system has been evaluated for X-, Y-, and Z-directions. In this experimentation, the workpiece object has been attached to the end-effector and a fixed point on the workpiece has been selected. Then, repeatedly, the robot manipulator is moved by one fixed step on a given axis, defining the relative error of the system as the difference between the displacement of a workpiece fixed point and the predefined step distance. This test has been conducted for 5 mm steps on the three main robot X-, Y-, and Z-axis. Figure 10 and Table 1 show the obtained results. As can be seen, the system provides similar errors for all axis and all tested directions, showing a maximum relative error of ± 2 mm. In addition, the system shows a stable performance, with an overall Euclidean error smaller than 2.2 mm in all directions.

5.2 Evaluation of system absolute error

Second, the absolute error of the system with respect to a manually defined ground truth has been evaluated. In this experiment, the generated system path for ten randomly located poses has been compared against a human defined ground truth, using a teach pendant. For each pose, 10 different tests have been conducted, with a total of 100 evaluations. The evaluation process is as follows: Initially, for a given random located object, the ground truth path, defined by four discrete distinguishable points, was manually found using the teach pendant. After that, the AOLP platform starts the object recognition module and the CAD model is properly located in the OLP platform, where the user can select the path. After the selection, the path is automatically generated from the CAD information and a simulation of the robot motion is performed, allowing the user to check the correctness of the trajectory. Finally, the automatically generated robot program is executed on the Denso industrial manipulator. Then, the system error is computed by comparing the captured ground truth reference points with the corresponding generated path. Figure 11 shows the whole process diagram.

The obtained results for each pose are presented in Table 2. The overall system results for all poses are presented in Table 3. Figure 12 shows the results for one test of an automatically obtained trajectory on simulation and real-world



(a) (b)

Fig. 9 **a** CAD model; **b** Kinect sensor data with CAD model recognized

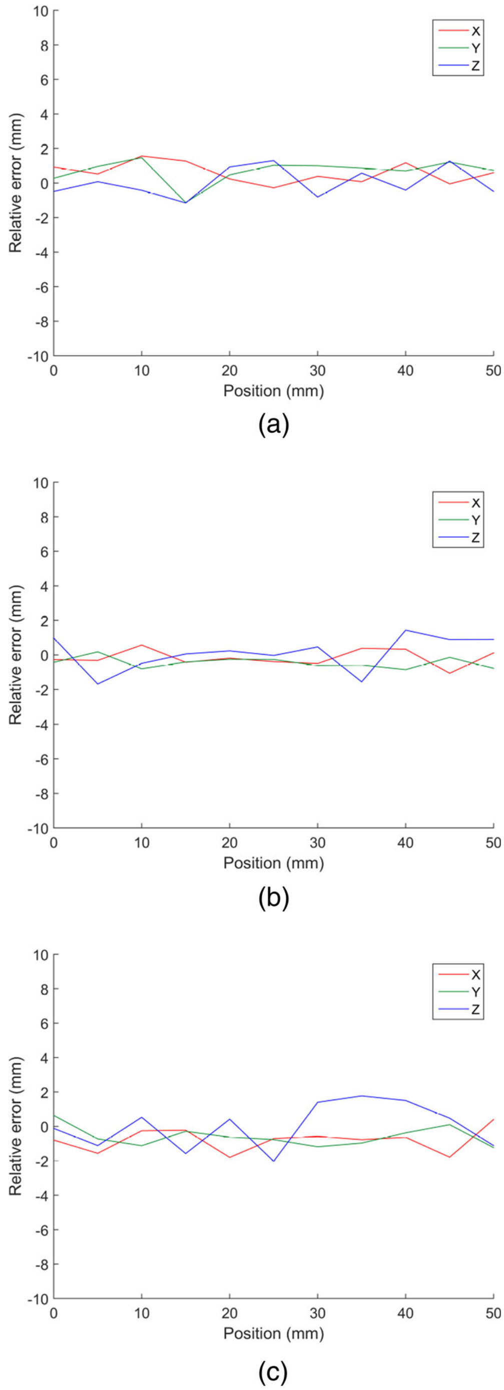


Fig. 10 Relative error of the system with respect to the industrial manipulator for 5 mm steps. **a** *X*-axis; **b** *Y*-axis; **c** *Z*-axis

scenario. Supporting previous experimentation, the obtained absolute error between the system path and human defined ground truth also shows similar results for all axis. Overall the systems show a mean positive error of around 2 mm with an std. deviation of 1 mm, for the *X*-, *Y*-, and *Z*-axis. Similarly, the systems show stable results for all different poses, obtaining consistent precision error for all cases. In this direction, the always positive overall consistent mean error of around 2.4 mm, ranging from 0.8 to 3.6 mm for different poses, can be probably attributed to the camera-to-robot calibration.

Table 1 Relative error with respect to the industrial manipulator for 5 mm steps on *X*, *Y*, and *Z* robot axis. Results in mm

Test	Mean error			Std. deviation		
Dir.	*X*	*Y*	*Z*	*X*	*Y*	*Z*
X	0.490	− 0.159	− 0.793	0.676	0.472	0.688
Y	0.690	− 0.450	− 0.595	0.688	0.318	0.581
Z	0.036	0.108	0.014	0.854	1.012	1.310

5.3 System robustness analysis

The performance and robustness of the three-dimensional vision system capabilities, in terms of recognition rate, rely solely on the vision module and has been already extensively analyzed on a comprehensive range of variate household and industrial objects for more than 60000 challenging test images in previous publications [28, 57]. These results, reaching recognition rates up to 100% for non-occluded cases, show the validity and robustness of the object recognition method for a wide range of different scenarios. In this sense, this characteristic has been widely evaluated on literature joining efforts from different authors, with solid and detailed knowledge available; therefore, no further tests have been conducted in this direction.

In another direction, focusing on the proposed integrated AOLP system performance for industrial manufacturing, we conduct a set of experiments to evaluate the precision performance of the integrated system against different illumination levels and different object materials. First, the system was tested under six different light conditions, from a highly illuminated environment to a completely dark scene, as shown in Fig. 13. In these experiments, no additional modification or parameter tuning has been used, following exactly the same procedures described in the previous section for the absolute error. Experimental results, presented in Fig 14, show the robustness of the proposed system for all different levels of illumination, obtaining consistent results with previous experiments. Although local minor variations occur, no critical drop of the system performance can be observed for different light conditions. These results show the validity of the proposed integrated system for working on different illumination environments within the same range of precision.

Second, the performance of the system for different object materials is evaluated. In this experiment, the proposed system precision is compared for four different

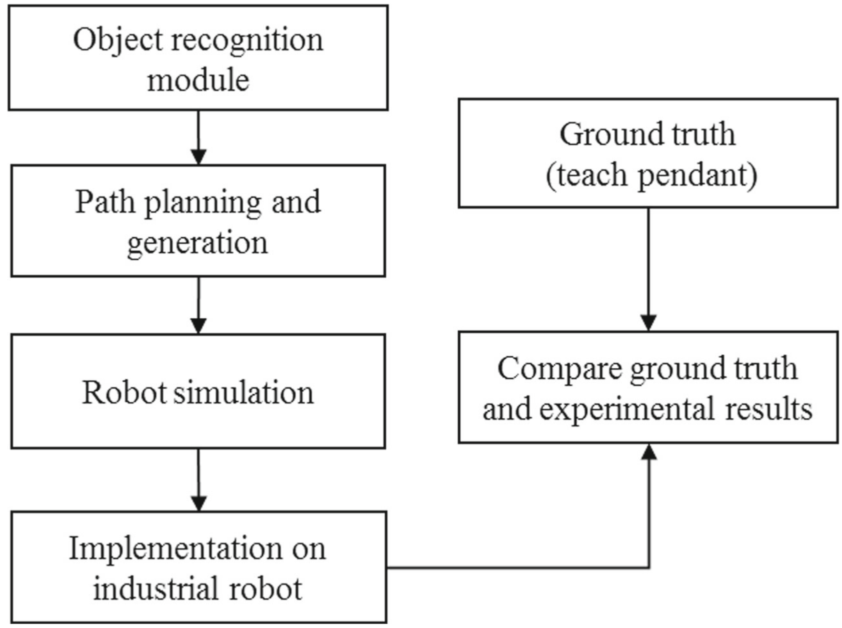


Fig. 11 Steps to compare the performance of the platform

objects made of foam, wood, metal, and plastic, as shown in Fig. 15. The objects where located on the same working position and tested on the same light conditions. Experimental results are presented in Fig. 16. As can be seen, the system shows robustness against all four different cases with noncritical minor variations between materials. Specifically, we can notice a slightly higher precision on wood and metal than plastic, which can be arguably attributed to their somehow smoother surface. In addition, the foam object shows an slightly higher error, which we attribute to the softness and nonrigidity of the object material. Overall, the obtained results show the robustness for different object materials with consistent precision.

Table 2 Absolute error per pose, with 10 tests per pose using 4 reference points. Results in mm

Pose	Mean error			Std. deviation		
	X	*Y*	*Z*	*X*	*Y*	*Z*
1	1.867	1.507	2.805	0.686	1.136	0.796
2	1.987	2.312	2.603	0.869	0.582	0.863
3	2.550	2.355	2.710	1.934	0.491	1.122
4	2.908	1.685	2.358	1.065	1.116	1.160
5	2.474	3.043	2.408	0.562	1.072	0.818
6	2.017	3.317	0.876	0.655	1.253	1.093
7	2.517	3.591	2.136	1.566	0.538	1.348
8	2.944	1.734	2.091	0.467	0.663	0.751
9	1.678	2.696	3.122	0.305	0.880	0.574
10	2.264	2.853	1.866	0.945	0.344	0.625

5.4 Comparison and discussion

Increasing demands of productivity on industrial manufacturing require the more precise definition of target workpieces and a higher level of control on manipulation and inspection processes. The challenges defined by tasks of different nature and high precision operations request novel automatic manufacturing approaches that can integrate benefits from several systems and techniques, defining innovative solutions to fulfill the requirements of those complex manufacturing processes. Some examples include integrated systems with cooperative robotics [15] for highly complex procedures, visual servoing techniques [36] for high precision tasks or novel 3D-based rendering techniques for automated optical inspection (AOI) [10]. These solutions could be employed to boost the productivity of still challenging industrial operations, such as insertion, high precision welding, 3D laser-cutting, and inspection of catastrophic failure and quality defects and dual arm robotized assembly. In this direction, most available OLP systems, relying on commercial platforms, lack of the necessary characteristics and flexibility required for this type of integration. Although an autonomous system has been proposed

Table 3 Overall absolute error for all poses, with 10 test per pose using 4 reference points. Results in mm

Mean error			Std. deviation		
X	*Y*	*Z*	*X*	*Y*	*Z*
2.320	2.509	2.297	1.025	1.055	1.055

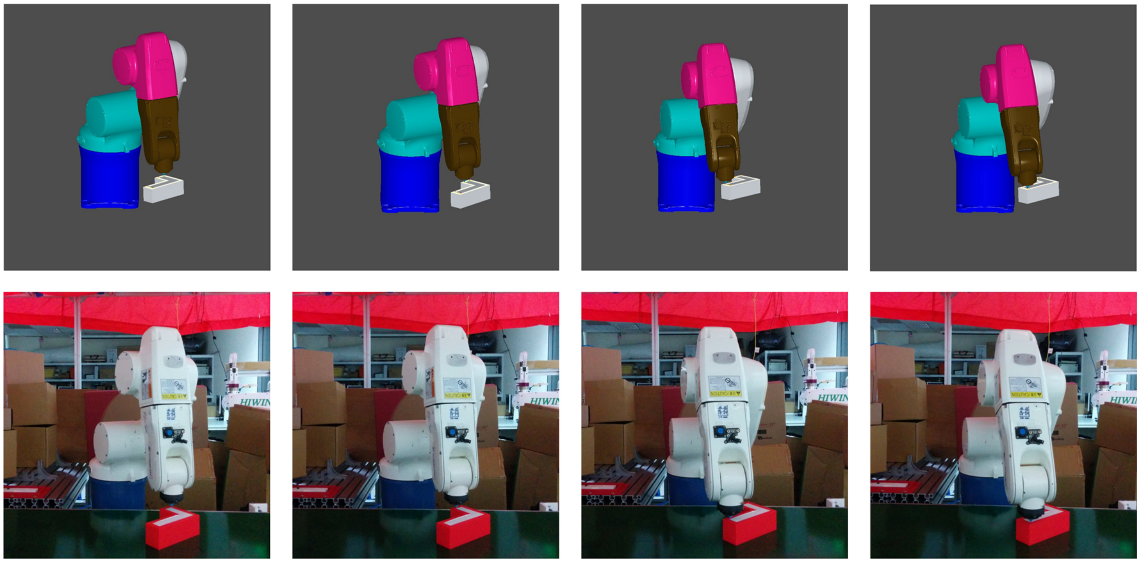


Fig. 12 Virtual and real-world results for one trajectory generated by the AOLP system

as an alternative and optimal solution, at present, their capabilities are limited to simple tasks, and they have not yet reached the necessary robustness required for complex manufacturing.

The presented solution proposes a novel approach that integrates the autonomous recognition capabilities of the three-dimensional vision systems with the user-friendly and workcell-free programming advantages from OLP platforms, defining a more productive and flexible solution. In this sense, the proposed system is based on a modular, independent and adaptable OLP platform, allowing to define a fully integrated architecture that can be coupled with other systems and extended its characteristics to face more challenging and complex manufacturing procedures. These characteristics can be in part archived by the implementation of the open CAD technology provided by

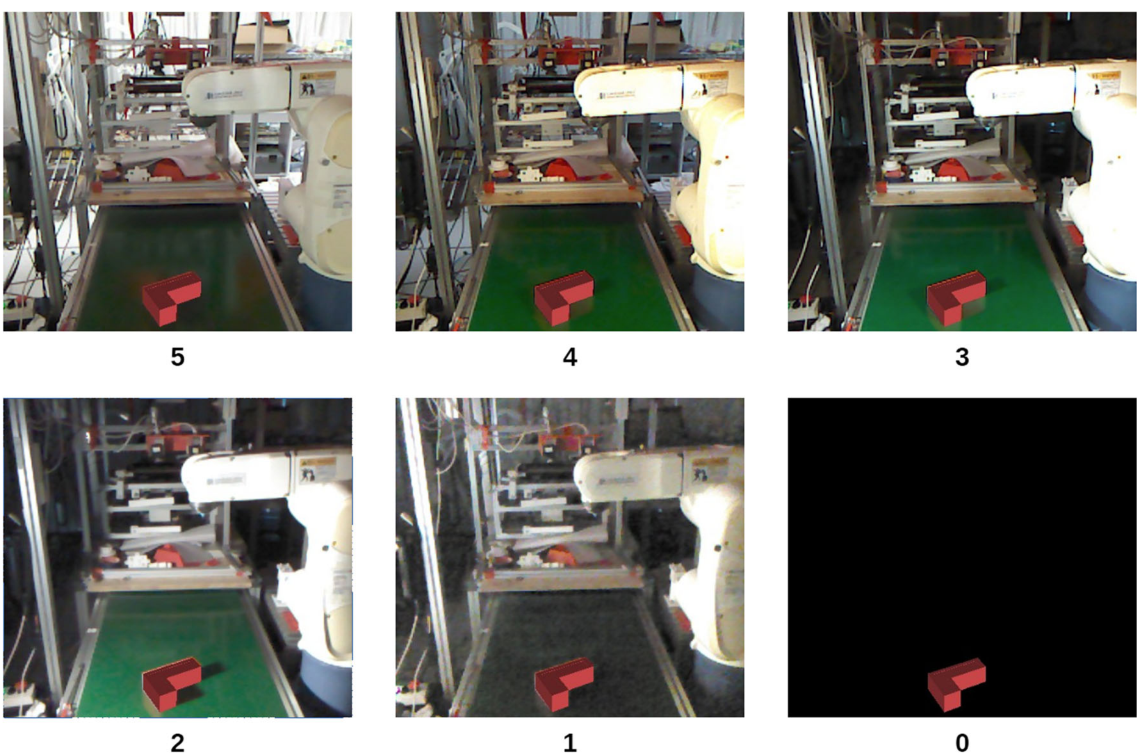


Fig. 13 Different tested scene illumination levels

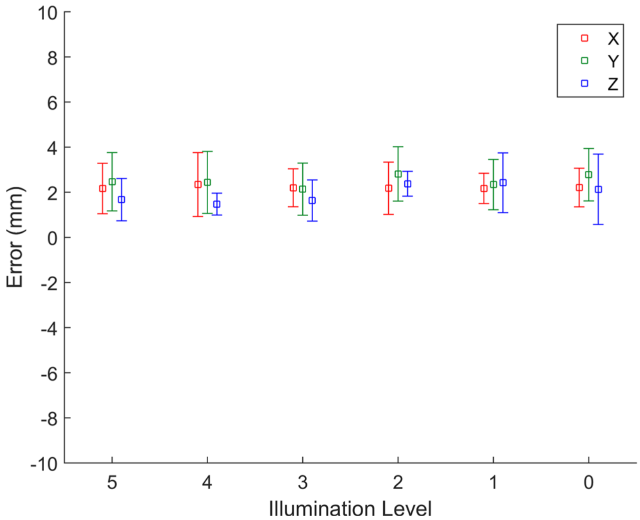


Fig. 14 System error for different illumination levels

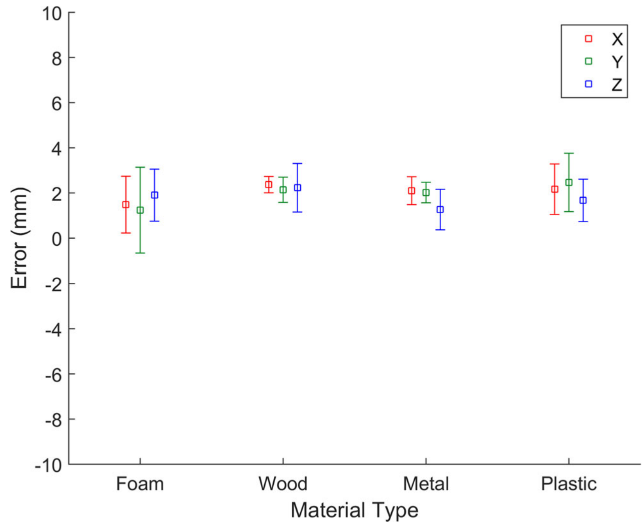


Fig. 16 System error for objects made by different types of material

Open Cascade. Therefore, the proposed system does not only join the autonomous workpiece detection capabilities of the three-dimensional vision but define a platform with an integrated architecture that can be applied and extended to a variety of directions and tasks, increasing productivity by means of a more efficient automatic robot integration.

The proposed method features are compared against other available state-of-the-art approaches for industrial manufacturing on 3D objects in Table 4. As can be seen, the proposed method combines most features while still providing a competitive precision. In this sense, solutions using custom automated path generation approaches are only designed to be applied to a limited range of cases. These specific solutions, usually costly and time-consuming to program, are based on constrained scenarios and are difficult to change for other purposes. In another direction, OLP methods provide tools to simplify the generation of robotic paths for all types of scenarios. However, these methods have a very limited range of automatic functions, rely usually on nonflexible commercial platforms and require a precise definition of the CAD environment, requiring costly human interventions and specific-purpose designs. In addition, their proposed architectures show limitations regarding their extension and integration into other systems. In order to overcome the problems related to the environment and workpiece definition, methods using object recognition can help to generalize automated approaches to various scenarios. The proposed combination of the OCC-based OLP platform with a state-of-the-art object recognition method by using a modular fully integrated architecture represents a compromise approach to most system limitations defining a more flexible and productive manufacturing solution. In addition, the proposed system has shown robustness in terms of vision recognition rates and overall system precision in different scenarios, showing their values for practical industrial applications.

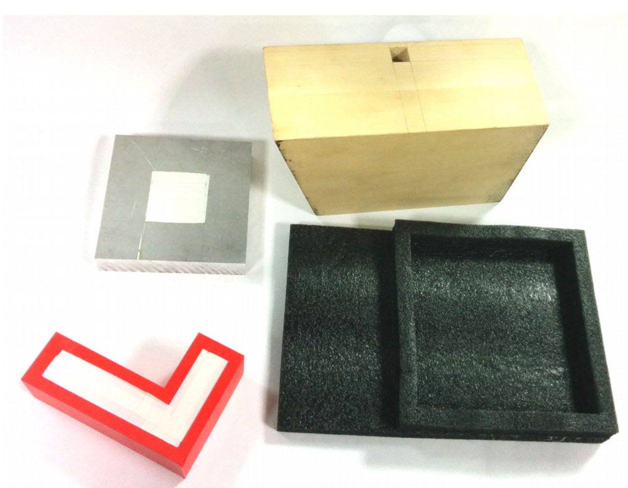


Fig. 15 Tested objects with different surface materials

Overall, the obtained results show the effectiveness and viability of the proposed system, showing consistent results in all experiments and tested scenarios, with a relatively good accuracy for low-demanding precision manipulation tasks. In addition, the flexibility of the integrated architecture allows the system to be easily coupled with more accurate 3D sensors, or integrated with other systems, such as visual servoing [36], to extend its functionality to automatic high precision tasks. On top of that, the intuitive and user-friendly platform allows the user to define the robot path, perform the simulation and generate the robot code with a few simple steps, defining a flexible solution for all types of requirements and manufacturing tasks.

Table 4 Comparison table between different methods' features for automatic industrial manufacturing on 3D objects

Method		Maiolino [39]	Caruso [16]	Neto [42]	Rocha [49]	Shah [52]	Our Method
OLP	Platform	RobotStudio Software	No	Autodesk Inventor API	No	No	OCC-based
	Auto path from CAD	Yes	–	No	–	–	Yes
	Robot code generation	Yes	–	Yes	–	–	Yes
	Commercial	Yes	–	Yes	–	–	No
Vision system		Kinect v1 Only depth	Kinect v2 RGB-D	No	Laser triangulation	Basler RGB	Kinect v1 Only depth
Path generation		OLP	Projected contour	OLP	No	Edge segmentation	OLP
Object recognition		SHOT/plane segmentation	No	No	SVM & Perfect match	No	Improved PPF
Architecture		Sofware UDP communication	Single module	Standalone Out-of-process	Single module	–	Modular Integration
Extension and Integration		Very limited UDP/add-on	Yes	Limited by OLP API	Yes	Yes	Yes
Experimentation		Yes	Yes	Yes	Yes	Yes	Yes
Shown robustness to		Illumination	Different objects	–	Different objects	–	Illumination, materials, clutter and diff. objects
System precision		–	< 1.5 mm	–	< 8 mm	< 7 mm	< 4 mm

6 Conclusion

A novel Automated Offline Programming (AOLP) system integrating the benefits of the three-dimensional vision systems and the OLP platforms has been proposed, developed and tested. The contribution of this work is the generation of a novel AOLP system with a fully integrated architecture joining an autonomous three-dimensional object recognition method and a flexible and intuitive OLP solution to overcome previous solutions' limitations. In detail, the proposed system provides a novel and unique combination of features that define a more productive system, overcoming the workpiece estimation challenges and architecture limitations of previous solutions. Furthermore, the proposed system has the capacity of integration and extension with other techniques to face high demanding precision operations and more complex manufacturing tasks. The system provides a fully integrated architecture for automatic robot programming, including workpiece pose estimation, path planning, simulation, code generation, and robot execution. In one hand, the system workpiece pose estimation relies on a highly robust and extensively tested three-dimensional vision method based on the point pair features voting approach. On the other hand, the proposed platform is a flexible and user-friendly solution that includes automated path planning from standard CAD models, which relies on OCC-based technology to allow an integrated architecture. The performance of the system has been analyzed under different scenarios by conducting four different experiments evaluating the system relative error, absolute error, and robustness against different illumination levels and object materials. The results of the experiments show the effectiveness and viability of the proposed approach, obtaining a consistent and relatively good precision for all cases. The system main characteristics have been compared with other available solutions, showing the strengths and unique combination of features of the proposed approach. In addition, the system has the capacity to improve its current features by using higher resolution sensors, extending its functions or integrating the system with other advanced techniques, defining more advanced solutions to accomplish highly demanding tasks. Overall, the proposed system represents a more flexible, cost-effective, and productive alternative to the existing approaches, representing a compromise solution between offline programming and autonomous manufacturing.

Funding information This work is financially supported by both Taiwan Building Technology Center and Center for Cyber-Physical System Innovation from the Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) in Taiwan. Additionally, this work was financially supported by the Ministry of Science and Technology, Taiwan (R.O.C), under the grant—105-2221-E-011-088-MY2.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. (2012). ROS-Industrial. https://rosindustrial.org/, accessed: 2018-09-10
2. (2017). OPEN CASCADE. www.opencascade.com, [Accessed 18-June-2018]
3. Sett A, Vollmann K (2002) Computer based robot training in a virtual environment. In: IEEE International conference on industrial technology, 2002, vol 2. IEEE ICIT '02, pp 1185–1189
4. Aldoma A, Vincze M, Blodow N, Gossow D, Gedikli S, Rusu RB, Bradski G (2011) CAD-model recognition and 6DOF pose estimation using 3D cues. In: 2011 IEEE international conference on computer vision workshops (ICCV Workshops), pp 585–592
5. Ames AL, Hinman-Sweeney EM, Sizemore JM (2005) Automated generation of weld path trajectories. In: (ISATP 2005). The 6th IEEE international symposium on assembly and task planning: from nano to macro assembly and manufacturing, 2005, pp 182–187
6. Andreopoulos A, Tsotsos JK (2013) 50 years of object recognition: directions forward. Comput Vis Image Underst 117(8):827–891
7. Baizid K, Cukovic S, Iqbal J, Yousnadj A, Chellali R, Meddahi A, Devedzic G, Ghionea I (2016) IRoSim: industrial robotics simulation design planning and optimization platform based on CAD and knowledgeware technologies. Robot Comput Integr Manuf 42:121–134
8. Bedaka AK, Lin CY (2017) Autonomous path generation platform for robot simulation. In: 2017 international conference on advanced robotics and intelligent systems (ARIS), pp 63–68
9. Bedaka AK, Lin CY (2018) CAD-based robot path planning and simulation using open cascade. Procedia Computer Science 133:779–785
10. Bedaka AK, Mahmoud AM, Lee SC, Lin CY (2018) Autonomous robot-guided inspection system based on offline programming and RGB-D model. Sensors, vol 18 (11)
11. Beetz M, Klank U, Kresse I, Maldonado A, Mösenlechner L, Pangercic D, Rühr T, Tenorth M (2011) Robotic roommates making pancakes. In: 2011 11th IEEE-RAS international conference on humanoid robots (Humanoids), IEEE, pp 529–536
12. Besl PJ, Mckay ND (1992) Method for registration of 3-D shapes. In: Sensor fusion IV: control paradigms and data structures, International society for optics and photonics, vol 1611, pp 586–607
13. Blanco JL (2010) A tutorial on se(3) transformation parameterizations and on-manifold optimization. Tech. Rep. 012010, University of Malaga
14. Brachmann E, Krull A, Michel F, Gumhold S, Shotton J, Rother C (2014) Learning 6d object pose estimation using 3d object coordinates. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T (eds) Computer vision – ECCV, vol 2014. Springer International Publishing, Cham, pp 536–551
15. Caccavale F, Uchiyama M (2008) Cooperative manipulators. Springer, Berlin, pp 701–718
16. Caruso L, Russo R, Savino S (2017) Microsoft kinect v2 vision system in a manufacturing application. Robot Comput Integr Manuf 48:174–181
17. Chen Y, Medioni G (1992) Object modelling by registration of multiple range images. Image Vis Comput 10(3):145–155
18. Correll N, Bekris KE, Berenson D, Brock O, Causo A, Hauser K, Okada K, Rodriguez A, Romano JM, Wurman PR (2018) Analysis and observations from the first amazon picking challenge. IEEE Trans Autom Sci Eng 15(1):172–188. https://doi.org/10.1109/TASE.2016.2600527
19. Daniilidis K (1999) Hand-eye calibration using dual quaternions. Int J Robot Res 18(3):286–298
20. Desouza GN, Kak AC (2002) Vision for mobile robot navigation: a survey. IEEE Trans Pattern Anal Mach Intell 24(2):237–267
21. Drost B, Ulrich M, Navab N, Ilic S (2010) Model globally, match locally: efficient and robust 3D object recognition. In: 2010 IEEE computer society conference on computer vision and pattern recognition, pp 998–1005
22. Eppner C, Höfer S, Jonschkowski R, Martín-Martín R, Sieverling A, Wall V, Brock O (2017) Lessons from the amazon picking challenge: four aspects of building robotic systems. In: Proceedings of the 26th international joint conference on artificial intelligence, IJCAI'17. AAAI Press, pp 4831–4835. http://dl.acm.org/citation.cfm?id=3171837.3171967
23. Everingham M, Eslami SMA, Van Gool L, Williams CKI, Winn J, Zisserman A (2015) The Pascal visual object classes challenge: a retrospective. Int J Comput Vis 111(1):98–136
24. Geng J (2011) Structured-light 3D surface imaging: a tutorial. Adv Opt Photon 3(2):128–160
25. Guo Y, Bennamoun M, Sohel F, Lu M, Wan J (2014) 3D object recognition in cluttered scenes with local surface features: a survey. IEEE Trans Pattern Anal Mach Intell 36(11):2270–2287
26. Hinterstoisser S, Holzer S, Cagniart C, Ilic S, Konolige K, Navab N, Lepetit V (2011) Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In: 2011 international conference on computer vision, pp 858–865
27. Hodan T, Zabulis X, Lourakis M, Obdrzalek S, Matas J (2015) Detection and fine 3D pose estimation of texture-less objects in RGB-D images. In: 2015 IEEE/RSJ international conference on intelligent robots and systems (IROS), pp 4421–4428
28. Hodan T, Michel F, Brachmann E, Kehl W, GlentBuch A, Kraft D, Drost B, Vidal J, Ihrke S, Zabulis X, Sahin C, Manhardt F, Tombari F, Kim TK, Matas J, Rother C (2018) Bop: benchmark for 6d object pose estimation. In: The European conference on computer vision (ECCV)
29. Horaud R, Dornaika F (1995) Hand-eye calibration. Int J Robot Res 14(3):195–210
30. Jing L, Fengshui J, En L (2016) RGB-D sensor-based auto path generation method for arc welding robot. In: 2016 Chinese control and decision conference (CCDC), pp 4390–4395, https://doi.org/10.1109/CCDC.2016.7531776
31. Kehl W, Milletari F, Tombari F, Ilic S, Navab NLeibe B, Matas J, Sebe N, Welling M (eds) (2016) Deep learning of local RGB-D patches for 3D object detection and 6D pose estimation, vol 2016. Springer International Publishing, Cham
32. Kehl W, Manhardt F, Tombari F, Ilic S, Navab N (2017) SSD-6D: making RGB-based 3D detection and 6D pose estimation great again. In: The IEEE international conference on computer vision (ICCV)
33. Kržič P, Pušavec F, Kopač J (2013) Kinematic constraints and offline programming in robotic machining applications. Tehnički vjesnik/Technical Gazette 20(1):117–124
34. Larkin N, Milojevic A, Pan Z, Polden J, Norrish J (2011) Offline programming for short batch robotic welding. In: 16th joining of materials (JOM) conference 2012, pp 1–6
35. Larkin N, Pan Z, Duin SV, Norrish J (2013) 3D mapping using a ToF camera for self programming an industrial robot. In: 2013 IEEE/ASME international conference on advanced intelligent mechatronics, pp 494–499
36. Lippiello V, Siciliano B, Villani L (2007) Position-based visual servoing in industrial multirobot cells using a hybrid camera configuration. IEEE Trans Robot 23(1):73–86

37. Mahmoud N, Ilhan EK (2016) Off-line nominal path generation of 6-DoF robotic manipulator for edge finishing and inspection processes. Int J Adv Manuf Technol 99(1-4):85–96
38. Maiolino P, Woolley RAJ, Popov A, Ratchev S (2016) Structural quality inspection based on a RGB-D sensor: supporting manual-to-automated assembly operations. SAE Int J Mater Manuf 9(1):12–15. http://www.jstor.org/stable/26268798
39. Maiolino P, Woolley R, Branson D, Benardos P, Popov A, Ratchev S (2017) Flexible robot sealant dispensing cell using RGB-D sensor and off-line programming. Robot Comput Integr Manuf 48:188–195
40. Malamas EN, Petrakis EG, Zervakis M, Petit L, Legat JD (2003) A survey on industrial vision systems, applications and tools. Image Vis Comput 21(2):171–188
41. Mitsi S, Bouzakis KD, Mansour G, Sagris D, Maliaris G (2004) Off-line programming of an industrial robot for manufacturing. Int J Adv Manuf Technol 26(3):262–267
42. Neto P, Mendes N (2013) Direct off-line robot programming via a common CAD package. Robot Auton Syst 61(8):896–910
43. Neto P, Mendes N, Araújo R, Pires JN, Moreira AP (2012) High-level robot programming based on CAD: dealing with unpredictable environments. Ind Robot Int J Robot Res Appl 39(3):294–303
44. Newman TS, Jain AK (1995) A survey of automated visual inspection. Comput Vis Image Underst 61(2):231–262
45. Pan Z, Polden J, Larkin N, Duin SV, Norrish J (2010) Recent progress on programming methods for industrial robots. In: ISR 2010 (41st international symposium on robotics) and ROBOTIK 2010 (6th German conference on robotics), pp 1–8
46. Polden J, Pan Z, Larkin N, Van Duin S, Norrish J (2011) Offline programming for a complex welding system using DELMIA automation. In: Chen SB, Fang G (eds) Robotic welding, intelligence and automation. Springer, Berlin, pp 341–349
47. Pollard NS, Hodgins JK, Riley MJ, Atkeson CG (2002) Adapting human motion for the control of a humanoid robot. In: Proceedings 2002 IEEE international conference on robotics and automation (Cat. No.02CH37292), vol 2, pp 1390–1397
48. Radovan H, Daynier RDS, Peter K, Roman R (2014) Offline programming of an ABB robot using imported CAD models in the RobotStudio software environment. Appl Mech Mater 693:62–67
49. Rocha LF, Ferreira M, Santos V, Moreira AP (2014) Object recognition and pose estimation for industrial applications: a cascade system. Robot Comput Integr Manuf 30(6):605–621
50. Rusu RB, Cousins S (2011) 3d is here: point cloud library (pcl). In: 2011 IEEE international conference on robotics and automation, pp 1–4
51. Rusu RB, Bradski G, Thibaux R, Hsu J (2010) Fast 3d recognition and pose using the viewpoint feature histogram. In: 2010 IEEE/RSJ international conference on intelligent robots and systems, pp 2155–2162
52. Shah HNM, Sulaiman M, Shukor AZ, Kamis Z (2018) An experiment of detection and localization in tooth saw shape for butt joint using KUKA welding robot. Int J Adv Manuf Technol 97(5):3153–3162
53. Shiu YC, Ahmad S (1989) Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form AX=XB. IEEE Trans Robot Autom 5(1):16–29
54. Tsai RY, Lenz RK (1989) A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. IEEE Trans Robot Autom 5(3):345–358
55. Tuytelaars T, Mikolajczyk K (2008) Local invariant feature detectors: a survey. Foundations and Trendsin Computer Graphics and Vision 3(3):177–280
56. Ulrich M, Steger C (2002) Performance evaluation of 2D object recognition techniques. Int Arch Photogramm Remote Sens Spat Inf Sci 34(3/A):368–374
57. Vidal J, Lin C, Martí R (2018) 6D pose estimation using an improved method based on point pair features, pp 405–409. https://doi.org/10.1109/ICCAR.2018.8384709
58. Wohlkinger W, Vincze M (2011) Ensemble of shape functions for 3d object classification. In: 2011 IEEE international conference on robotics and biomimetics, pp 2987–2992
59. Zhu W, Qu W, Cao L, Yang D, Ke Y (2013) An off-line programming system for robotic drilling in aerospace manufacturing. Int J Adv Manuf Technol 68(9):2535–2545