OBJECT RECOGNITION AND USER INTERFACE DESIGN FOR VISION-BASED AUTONOMOUS ROBOTIC GRASPING POINT DETERMINATION

FANG-CHE CHENG[1], CHIA-CHING YEN[2] and
TAY-SHENG JENG[3]
[1,2,3]*Department of Architecture, National Cheng Kung University, Taiwan*
[1]*fccheng0902@gmail.com* [2]*ccyen@umich.edu*
[3]*tsjeng@mail.ncku.edu.tw*

Abstract. The integration of Robot Operating System (ROS) with Human-Machine Collaboration (HMC) currently represents the future tendency toward Autonomous Robotic In-Situ Assembly on Construction Sites. In comparison with the industrial environment, construction sites nowadays are extremely complex and unpredictable, due to the different building components and customized design.This paper presents a visual-based object recognition method and user interface enabling on-site robot arms to autonomously handle building components, to build specific designs without the influence of material, shape, and environment. The implementation is an object recognition approach that serves with KUKA industrial robotic manipulator along with an RGB-depth stereo camera in an eye-in-hand configuration to grasp and manipulate found elements to build the desired structure. Opportunities for using vision-based autonomous robotic in-situ assembly on construction sites are reviewed.

Keywords. Computer vision; robot operating system; object recognition; pose estimate; grasping point determination; human-robot collaboration.

1. Introduction

In the field of architecture and engineering, Robotic fabrication has been the predominant research subject recently. Robotic autonomous stands as a promising field for construction, it provides an efficient, more secure, and more precise way for processing architecture on-site resources, which allows extending more possibilities of creative and customizable architecture. While the autonomous robotic assembly has great benefits on the construction process definitely, there are still limits on the application in the projects of real construction. On-site robotic construction differs from robotics in the manufacturing industry should be taken into account the evolving, massive and unstructured environment of typical work sites. It is difficult to make robots adapt to the continually changing site without actively participating in the building process. The growing accessibility

PROJECTIONS, Proceedings of the 26th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA) 2021, Volume 1, 633-642.

of closed-loop robotic fabrication, promises the industry a way to automate aspects of the construction process, either on- or off-site, that require feedback from the physical environment for their success (Tish, King, and Cote, 2020)

1.1. PRECEDENTS

Although the use of industrial robotic arms can improve the adaptability and accuracy of the automated transformation system, the traditional robotic arm processing path planning requires tedious and time-consuming point-to-point teaching through the teach pendant, so that the robotic arm can perform processing operations according to the planned path. Moreover, the operator usually does not have the professional skills in this area and must go through a prolonged period of training to use the teaching pendant to plan the processing path of the industrial robotic arm. More importantly, whenever the processing path needs to be re-planned, the entire production line will be interrupted as a result, causing losses to the manufacturer.

As the real construction environments are normally chaotic and unstructured, the posture of the object to be processed in the environment appears randomly, only using CAD-system to plan the path of the robotic arm cannot process objects with random poses. Therefore, it is no doubt that construction robotics applications need computer vision and other sensing mechanisms to supply the robot with real-time data and information of its physical environments. Giftthaler et al. presented that the stereo camera rig fixed on the base of the robot observes was can be tailored to very different sensing tasks by modifying their configuration and the subsequent image processing, cameras are the sensing modality with the highest potential for on-site robotic building construction. (Giftthaler et al. 2017). Feng et al. also presented vision-guided autonomous robotic assembly research that addresses these challenges and enables autonomous robotic assembly of freeform modular structures on construction sites (Feng et al. 2015). Other research has also show computer vision used to fabricating and assembling irregular components is controlled by sensor-enabled material selection. (Wu and Kilian 2016). In many of the projects described above, the computer vision system perceives the information of the working environment through the image sensor, thereby processing and analyzing the environmental information, allowing the robotics to autonomously complete more complex processing operations.

2. Related work

2.1. COMPUTER VISION

In the construction industry, computer vision has immense potential. The computer vision system was developed to detect the targets and estimate the robotic pose of unmarked construction objects in the work cell. The end-effector is equipped with an Intel RealSense D435 RGB-Depth camera. An active infrared stereo is used for the camera to build three-dimensional point clouds of its field of view and also includes an RGB color sensor. The depth accuracy of the D435 cameras is less than one percent of the distance from the object. The accuracy is about 2.0mm to 5.0mm if the D435 camera is located 1 meter above the object.

This camera provides a great field of view of approximately 10 meters, along with a global shutter on the depth sensor that is ideal for fast-moving applications and computer vision.

2.2. COORDINATE SYSTEM TRANSFORMATION AND HAND-EYE CALIBRATION

In computer vision with the robot arm, the hand-eye calibration problem (also called the robot-world calibration or robot-sensor problem) is an important issue of determining the transformation with a camera and end-effector or the robot base of the world coordinate system. The use of a camera in a robot control loop can be performed with two types of architecture: the camera is said eye-in-hand when rigidly mounted on the robot end-effector and it is said eye-to-hand when it observes the robot within its workspace (Fig. 1). These two schemes have technical differences and they can play complementary parts. The eye-in-hand one has a partial but precise sight of the scene whereas the eye-to-hand camera has a less precise but global sight of it (Gregory Flandin, Francois Chaumette & Eric Marchand, 2000). To allow the position of the camera to be flexibly changed, and to avoid the problem of the camera being blocked by the robotic arm, the configuration with eyes-in-hand is performed here.

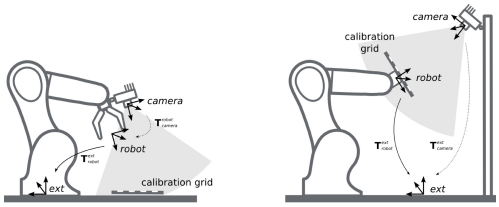

Figure 1. Two situations with hand (robot) eye (camera) calibration. Eye-in-Hand Calibration and Eye-to-Hand Calibration.

This paper, using the OpenCV library method to find the conversion matrix between the 2D image and the 2D coordinates of the robot tool. In the beginning, a color image containing a calibration chessboard is acquired through the camera, and then the robotic arm is manually controlled to locate the four corners of the board to record the results in sequence. Using OpenCV functions to find the actual positions of the four corners on the chessboard in the image (Fig. 2). With the coordinate position of the robotic arm tool PT and the position of the corner points of the image PC, it is necessary to find a 3x3 conversion matrix M and substitute it into the equation. Set Z value in the space as 1, and input any two-dimensional coordinate values to acquire the coordinate of the robot arm tool after matrices are multiplied. The formulas are as follow:

$$P_T = MP_C \tag{1}$$

$$\begin{bmatrix} p_1^x & p_1^y & 1 \\ p_2^x & p_2^y & 1 \\ p_3^x & p_3^y & 1 \end{bmatrix}_T = M \begin{bmatrix} p_1^x & p_1^y & 1 \\ p_2^x & p_2^y & 1 \\ p_3^x & p_3^y & 1 \end{bmatrix}_C \tag{2}$$

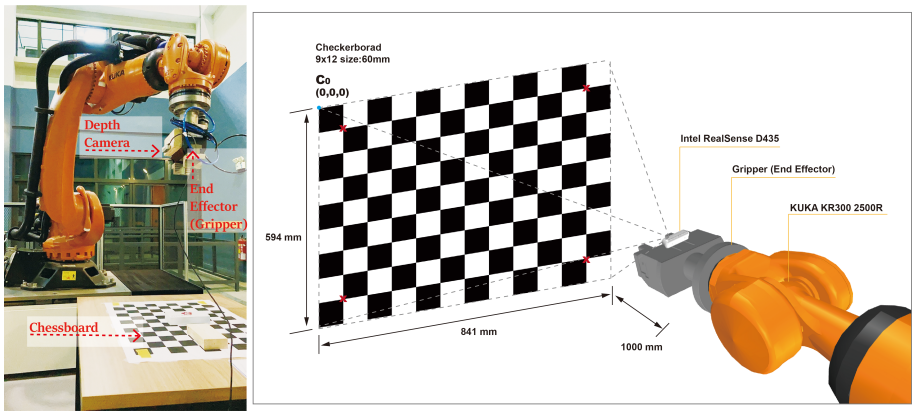

Figure 2. Combination of the robot arm and depth camera. Eye-in-hand calibration with depth camera and robot, and locate four corners of the Cali-chessboard.

This calibration method is more robust and easier to collaborate with different end effectors, the approach can be applied to the whole coordinates of workspace viewed through the lens to spatialize data in the coordinate system of the robot.

2.3. OBJECT RECOGNITION

Compared to the industrial environments, the construction sites are much more complex and unstable. There are two approaches to present as the object recognition system was developed: RGB-based object recognition and Depth image object recognition. According to the results, to make the object recognition system more stable and reduce the interference effect generated by the external environment, this recognition method discards RGB recognition and adopts depth recognition (Fig.3). As all of the targets and environment workspace were placed, the camera captures a 1280 x 720 resolution depth image above the platform 90° angle of depression and sets this position as a visual point. First, perform a threshold operation based on the depth value in the image to obtain the mask of the object -this method to get the mask is faster and more accurate. Second, blur the mask to remove noise and smooth the object boundaries of the mask. Third, introduce morphological closing operation to obtuse acute angles of the image, and maintain background areas that have a similar shape to this structuring characteristic. Last, detecting the shape of each object contours through open CV libraries, and check to determine if the approximate object in view. This depth recognition method allows the robot's detection system to solve the problem that objects may fail to detect when the object is in the same color background or due to the influence of light, and it also allows the program to detect targets of different colors or materials.

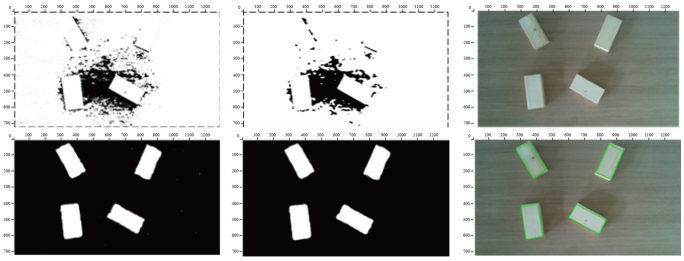

Figure 3. Comparison results of depth recognition and RGB recognition. The images above are HSV hue thresholding, and the bottom row is the results of depth image thresholding.

3. Autonomous robotic grasping system

3.1. SYSTEM CONTROL FLOW

To put the system into a construction application, this research developed an autonomous robotic recognition system (ARR), which can detect the design components and their properties without placing the robot in a controlled environment. Allowing it to automatically look for every available object in the view or raw materials, and it autonomously estimates the robot posture and target angle for the end effector to grasp. The user runs it on an online computer and is written in Python and reduces the requirements for control panel operations.

The Framework of the ARR comprises two systems, the object detection system and a grasping pose estimation system (Fig.4). Both systems depend on computer vision to work; first, the object detection system captures one RGB and one depth image from the working environment through the depth camera Intel RealSense D435. Thresholding the depth image to get the object's mask, then blur it to remove noise into smooth boundaries. The final output results through a closing operation to obtain the object image and contour information. Second, the grasping pose estimation system calculates the rotation angle and center point of the object through the four corners of the contour, and input the pick and place position to avoid collision between robot and objects.

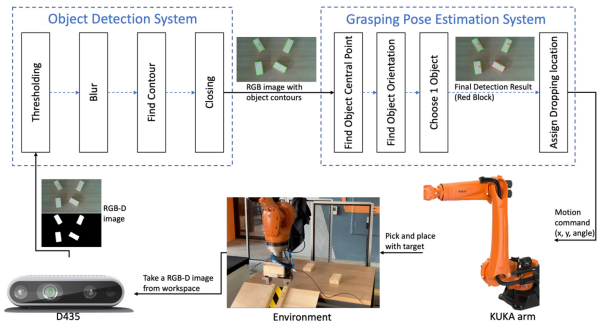

Figure 4. Closed-loop control flow diagram of the autonomous robot object recognition system.

As all the system data is completely collected, a sequence of motion outputs can be obtained. The robot arm processes the given action outputs in sequence and completes the grasping task in the working routine. Finally, we let the camera take images from the environment again and detect whether there are objects on the screen. If the environment is empty, stop the routine; otherwise, the system acquires a newly RGB-D image as the output of the object detection system and repeats the above steps until all objects in the working environment have been grasped to stop.

In the following text, there are more insight details about the object recognition algorithm to introduce the information with how it presents, which is helpful to understand the limitations and potentials of the system later.

3.2. POSE AND ANGLE ESTIMATION

With the contour frame of object identification now in hand, the next step is to estimate the grasping pose of the robotic arm. Thus, a motion planning algorithm was developed to find the optimal grasping position and plan a collision-free motion to deliver the target. The motion planning algorithm was based on the following:

(A) Object central pointMove the lens to the position of 90° angle of depression above the platform, using the contour of the object to find the four corner points of each, and use the average to calculate the center point of the object (Fig.5).(B) Angle of the detected objectSeeing as the object to be clamped in the experiment is a square, the vector of the long side and the short side can be calculated through the four corner points, and the long side vector is selected as the rotation angle of the object (Fig.5).(E) End effector rotationThus, the system generates the rotation angle of each target based on the vectors, this range of values is between -180 degrees and 180 degrees (Fig.5). (D) Grasping selectionAfter calculating the gripping pose estimations of all objects, we select the object with the lowest Y-axis position in the screen as the most preferred object to be gripped. The initially picked object is displayed in a red frame, and other non-target objects are displayed in a green frame.(E) Robot movementBy feeding the gripping posture and drop location of the target, we simulate robot behavior for each specified sequence of actions on all objects. Besides, we generated a plug-in for KUKA PRC in grasshopper to simulated and visualized robot path planning.

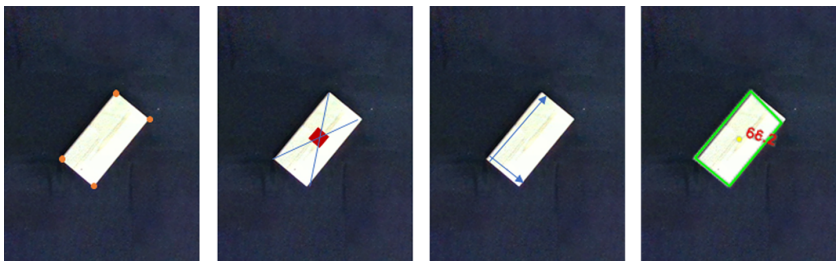

Figure 5. Object central point and angle estimate process.

3.3. HOME AND SYSTEM ROUTINE

For the robot to perform the assembly task, a list of sequential commands is generated as follows: (1) Positioning the camera above the workspace to capture the entire field of view and set the position to home Ph. Calculate the transformation matrix between the robot and camera based on a calibration chessboard. (2) The object detection system checks every object that is present in the view. (3) To avoid collision between robot and object, move the end effector to a position about 500 mm above the object. (4) Move slowly to target and rotate the end effector to the grasping through the angle and pose estimate algorithm. (5) Pickup and place the target to assembly. (6) Move back to Ph along the generated path, and the camera runs the object recognition routine (Fig.6).

After one object is placed, the control algorithm loops return to the robot's Home state to re-estimating object recognition, there are two reasons for this: first, there may be errors to use a single pose estimation for the robot arm to complete all the grasping postures. If the working environment changes, for instance, different numbers of objects or workspace, then the robot arm performs the grasping action according to the original detection result. However, this error may be corrected once the system re-estimates the grasping posture in every home state. Moreover, if a single detection failure results in the omission of some objects, then the system ignores it if the pose estimation is not performed again, making the grasping task incomplete. Another great advantage of re-estimating the pose is that the execution process can interact with the environment, such as manually adjusting the position of objects, removing or adding different ones, making the entire recognition process more flexible. In the meantime, we accomplished a pick and place test of the object recognition routine in Robot Aided Creation and Construction (RAC-Coon) of National Cheng Kung University.

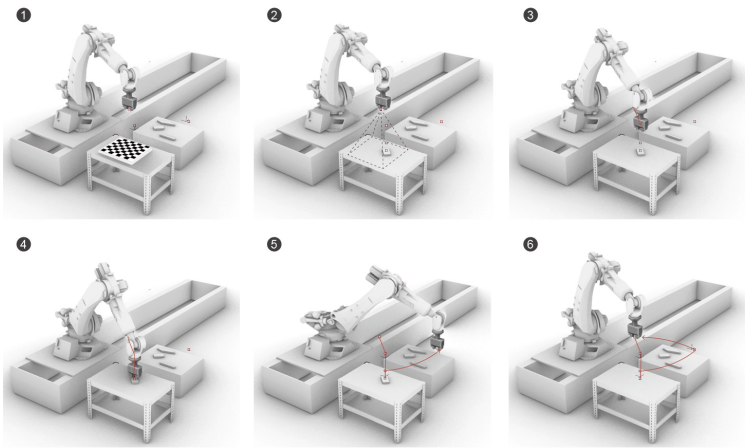

Figure 6. Visual-based autonomous pick and place workflow: 1. Home set and hand-eye calibration (this step only set at the beginning), 2. Object recognition and pose estimate (system routine), 3. Grasping position, 4. End effector angle and pick the target, 5. Place and assemble, 6. The robot moves to the home state and detects the environment again.

Figure 7. Images of pickup and feed to ramp process for the autonomous robotic recognition system in the laboratory.

4. Human-Robot Collaboration interface

To put the system into fabrication application, an autonomous construction workflow interface was designed, and then all the real-time information of the shapes and spatial properties are imported to a Human-Robot Collaboration (HRC) system. Allowing it to automatically find every target in the working environment either materials or shapes, and it can also achieve Computer in the Human Interaction Loop (CHIL) to choose the desired object.

The framework of the HRC interface consists of two systems, automatic object recognition, and a user manually control process (Fig.7). Both systems depend on this object recognition system to work. The automatic object recognition requires an environment image from the Intel RealSense d435 camera that is then displayed on the left screen, the screen will be updated in real-time during program execution. The screen on the right is the recognition result image, which includes the central point and pose angle, the depth recognition program automatically outputs the result as detected in every movement. The manual planner process allows the user to manually click on the selected object in the real-time screen, and then the robot moves to the specified position to grasp the target. This system not only provides user control of the workflow, but it also can test the program of the calibration results. The buttons in the bottom row are used to control the movement of the robotic arm in three-dimensional space and the on/off functions of the gripper through the program. Also, there is a “grasping” button, when the button is triggered, it can automatically detect objects on the screen and use this system to estimate the pose of the grasping. There is a text output function at the top of the screen, which describes what action the program is performing in the current state so that users can track the movement immediately.

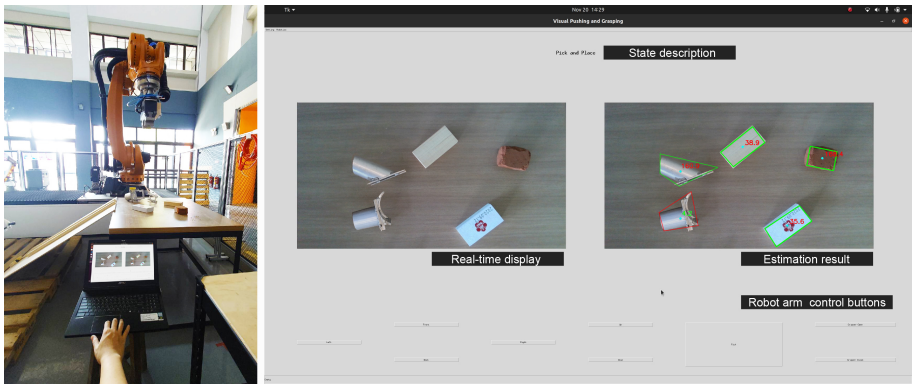

Figure 8. Image of human-robot collaboration and user interface.

5. Result and Discussion

To realize the possibility of intelligent construction in the future, the research here presents a highly accessible approach of open-source libraries and low-cost components. These open-source libraries can provide assembly processes with more stable and vision-guided robot fabrication for the construction industry. This system reduces some requirements for professional training in industrial robot control and can apply to off-site construction or even potential for on-site construction.

However, more tasks could be solved with robotic computer vision to improve the fabrication process and complex application. The object recognition system is developed based on 2D depth images; thus, it cannot effectively recognize 3D object information and activate the robot to complete more difficult assembly. In the real construction site, multiple spatial point clouds are required to be registered and edited into the recognition system. An effective way is to introduce additional cameras into the system to estimate the posture evaluation and angle results between the object and the robot in the three-dimensional space, it can provide more accurate targeting to increase the precision around. In addition, the integration of object recognition with the BIM models can improve the performance of robotic fabrication. In this framework, the BIM models of the project are the main base to drive the autonomous system, each of information on the building components can be provided and searched easily, and more design and requirements about the assembly process can be added in future experiments. It helps to record the details and tries to plan more effectively on-site for robot-to-construction.

6. Conclusion

Industrial robots are excellent at repetitive tasks in the controlled surrounding, but the opposite is true on construction sites. In-situ robots indeed require layers of sensing and intelligence to be able to adapt to the unstructured working environment and real-time variability with few to no reprogramming. The object

recognition developed in the research makes for robotic assembly construction processes to bring advantages with convenience, flexibility, and affordable recognition.

This research develops eye-in-hand coordinate spatial transformations, object recognition, and user interface required for pick and place assembly close-loop workflows through the depth-vision camera, integrating the object detection system and grasping pose estimation system into a Human-Robot Collaboration system for real-time robotic control. This object recognition system is constructed with Open-Source Computer Vision Libraries and low-cost components, permitting the accessible technology deployment in several applications.

It shows the great potential of the proposed system to be operated as an intelligent, sustainable, and efficient approach for on-site autonomous applications. Moreover, it allows more possibilities for robotic fabrication to achieve more complex and special design, exploring a new collaboration between humans and robots in the intelligent construction area.

Acknowledgments

This research is developed by Robot Aided Creation and Construction (RAC-Coon) of National Cheng Kung University and the Department of Architecture. We would like to thank Dr.Yi-Hsuan Tu provides the resource of computer vision and support, Dr. Yang-Ting Shen for advice on the human-computer collaboration, Dr. Sheng-Fen Chien for advice on the pose estimation, Ying-Syu Huang for providing the robotic coding, and Yu-Xun Chen for providing the method of calibration and coding.

References

Chuang, S.: 2019, *Picking of Stacked Workpieces by Robot Arm Based on RGB-D images*, Master's Thesis, National Chung Cheng University.

Dawod, M. and Hanna, S.: 2019, BIM-assisted object recognition for the on-site autonomous robotic assembly of discrete structures, *Construction Robotics*, **3**, 69–81.

Feng, C., Xiao, Y., Willette, A. and Kamat, V.R.: 2015, Vision guided autonomous robotic assembly and as-built scanning on unstructured construction sites, *Automation in Construction*, **59**, 123-138.

Flandin, G., Chaumette, F. and Marchand, E.: 2000, Eye-in-hand / Eye-to-hand Cooperation for Visual Servoing, *Proceedings - IEEE International Conference on Robotics and Automation 2000*, San Francisco, 2741-2746.

Giftthaler, M., Sandy, T., Dörfler, K., Brooks, I., Buckingham, M., Rey, G., Kohler, M., Gramazio, F. and Buchli, J.: 2017, Mobile robotic fabrication at 1:1 scale: the in situ fabricator. Constr Robot, *Construction Robotics*, **1**, 3-14.

Tish, D., King, N. and Cote, N.: 2020, Highly accessible platform technologies for vision-guided, closed-loop robotic assembly of unitized enclosure systems, *Construction Robotics*, **4**, 19-29.

Wu, C., Jiang, S. and Song, K.: 2015, CAD-based pose estimation for random bin-picking of multiple objects using a RGB-D camera, *Proceedings of ICCAS 2015*, Busan, South Korea, 1645-1649.

Wu, K. and Kilian, A.: 2016, Developing architectural geometry through robotic assembly and material sensing., *Robotic Fabrication in Architecture, Art and Design 2016*, Sydney, 240–249.