Autonomous brick picking with a mobile manipulator robot

Ioannis Papakis, Erik Komendera *

***Abstract*— In this draft paper, a robotic platform is used capable of locating a model brick, namely a structure element, and picking it up. The brick is assumed to be within reach of the manipulator without requiring the base to move in our assumption. The problem is tackled in two steps, locating the brick's configuration in 3D space and positioning the robotic manipulator in place for pickup operation.**

I. Introduction

Autonomous robotic interaction with the environment has been a major research topic in robotics over the last decades. With collaborative robots appearance [1] [2], soft robotic manipulators, on site mobile robots [3], their use in laboratory and industrial environments is increasing [4] not only in numbers but in much more intelligent and higher level autonomy.

The motivation behind this paper, is Challenge Two of the MBZIRC 2020 international robotics competition in Abu Dhabi [5]. Challenge Two, is based on autonomous building of a brick wall, using UGVs and UAVs. The goals, beyond that particular challenge of the competition, are to develop technologies for autonomous operation of robots that involves interaction with the environment, multi-robot coordination and assembly planning.

Within that scope, is the objective of this paper. At first, a brief description of similar work on the subject of assembling a structure is made. Then, we split the task to the specific problem areas, presenting the methodology and the experimental results with discussion.

II. Related Work

In robotic applications that require assembling a structure, many new innovative systems have been developed that have scene understanding and some level of autonomy. As described here [6], industrial systems such as the Bronco robot or SAM(Semi Automated Mason) are existing bricklaying robots. The Kuka Moiros is another one that handles welding processes.

There are also experimental robots in laboratory environments, making the distinction here between aerial and ground vehicles. Ground platforms such as dimRob [7], a stationary ABB manipulator equipped with a 3D laser scanner, Termes a climbing building robot [8], marXbot a building robot [9] localized by a Vicon, motion tracking system, have been presented.

Other aerial vehicles, such as the four quadrotors which built a wall structure at ETH Zurich using the VICON system have been successfully tested [10]. The main existing problems associated with these systems is the duration that they can operate as well as the lifting capacity.

These and other solutions have proven that precise positioning is necessary, especially in conditions where the environment is dynamic or the assembly task requires precision and is not trivial such as self-aligning blocks.

III. Approach

A. Problem Statement

The process of building a wall of model bricks, can be split into many different stages. These include the robot being far from the targets, having detected them or not, but also the robot being within reachable grab radius. The robot can also be moving or be stationary in each of these stages correspondingly.

In our scenario, the problem in concern is the autonomous picking of model bricks. The robot in use, is a mobile robot with a manipulator mounted on board. It has to be able to perform the visual detection, locate the end-goal position and do the motion planning task to achieve that goal. The mobile robot remains stationary in that stage.

B. Methodology

The approach followed, is a three stage approach in which robot detects the location of the brick, moves the manipulator into the end goal position and orientation and finally lifts the brick.

The model bricks used, are custom made cardboard boxes. These boxes have been painted and a metal plate is attached on their top surface along with an AR code above. The system in use, is the mobile platform Husky from Clearpath Robotics. The robotic manipulator is Sawyer from Rethink Robotics. Other system hardware includes the ZED camera, as a fixed camera system and the Sawyer onboard end-effector camera. The end-effector also has as attachment an electro-permanent magnet.

1) Object detection and tracking: For the first stage of the approach, two different systems were tested. The first one involves a camera at the base of the robot. The other one uses a camera at the end effector frame.

For the first method, the stereo vision ZED camera was used. Initial tries included the detection of the brick as an object without the use of a code. That technique used the SIFT method for object detection by applying the find object 2d package [11] . Other methods used with the ZED camera

*The purpose of this draft paper is to fulfill the requirements for the Robotic Interaction, Manipulation and Assembly class taught by Professor Eric Komendera of the Mechanical Engineering Department at Virginia Tech, USA.

system involved the detection of a QR and AR code on top of the brick, using the visp auto tracker [12] and ar track alvar [13] packages respectively.

As a second approach, the same detection packages were used but this time the end-effector camera was used. This camera is looking at the brick from above as in figure 1b, thus this time the top surface is parallel to the camera frame in comparison with the previous base camera, in which the two frames are at an angle.

The find object 2d package is able to return a 3D position of the object in the camera frame, however the location is not on the top surface but rather at the center of the object we are trying to locate. The other two return the 3D position of the center of the code.

For the filtering of that position, a plain Kalman filter is used for the purpose of incorporating robustness in the measurements. The system receives the final position of the object after a chosen elapsed time currently at 1.5 sec, having filtered the measurements during that time. Since both target and robot are static, the operation of the filter done is the following:

$$x_{k|k} = x_{k|k-1} + K_k y_k$$

$$P_{k|k} = (I - K_k H_k) P_{k|k-1} (I - K_k H_k)^T + K_k R_k K_k^T$$

where

$$y_k = z_k - H_k x_{k|k}$$

$$S_k = R_k + H_k * P_{k|k-1} H_k^T$$

$$K_k = P_{k|k-1} H_k^T S_k^{-1}$$

After observing the system measurement variances, the initial P and R values are set for the task.

2) Manipulator planning: To achieve the positioning of the arm at the end-goal, the Intera-SDK environment was used. This is used by developers for custom programming of the Sawyer robot. The main methods implemented as shown in figure 2, include the position of object from camera to base frame transformation, the calculation of the inverse kinematics for the seven joints of the robot, the motion trajectory generation and the impedance allowance option for the joints.

The camera to base frame transformation uses the following scheme for the on rover ZED camera:

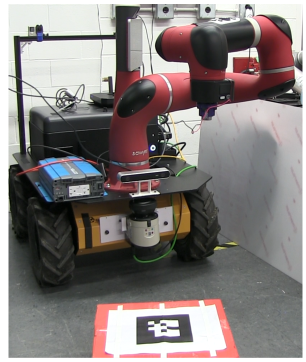
$$^{c_r}X_r = Rot_z(90°) * Rot_y(90°) * T(-0.12, 0, 0) * {}^{c_r}X_o,$$

where $^{c_r}X_r$ is the position in the robot frame, $^{c_r}X_o$ the position in the camera frame on the rover and for the robotic arm camera:

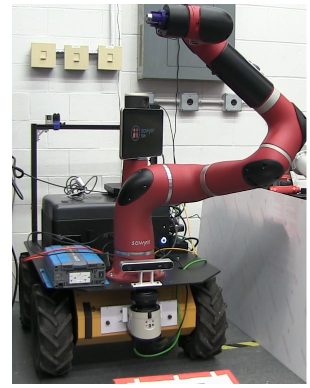
$$^{c_e}X_r = Rot_x(90°) * Rot_z(90°) * T(0.4, 0, 0.6) * {}^{c_e}X_o,$$

where $^{c_e}X_r$ is the position in the end effector camera frame.

For the inverse kinematics, the calculation is done in a numerical approximation sense, since we have a 7 DOF robot, thus an analytical solution is not performed. The final



(a) ZED camera



(b) Manipulator camera

Fig. 1: The camera system used and manipulator orientation before picking operation for each case

position of the object expressed as $X_o = [x, y, z]^T$, is used in the expression of the Jacobian

$$J = \begin{bmatrix} \frac{\partial X_0}{\partial q_1} & \dots \frac{\partial X_0}{\partial q_7} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial q_1} & \dots \frac{\partial x}{\partial q_7} \\ \frac{\partial y}{\partial q_1} & \dots \frac{\partial y}{\partial q_7} \\ \frac{\partial z}{\partial q_1} & \dots \frac{\partial z}{\partial q_7} \end{bmatrix}$$

The function x=f(θ) is defined as the forward kinematics function and $g(\theta) = X_r - f(\theta) = 0$. With an initial guess θ_0, θ_d can be finally estimated after consecutive approximations from the Taylor expansion $x_d = f(\theta_d) = f(\theta_0) + \frac{\partial f}{\partial \theta} |_{\theta^0} *(\theta_d - \theta^0)$, where $\frac{\partial f}{\partial \theta} |_{\theta^0}$ is the Jacobian previously described. In the implementation the inverse kinematics solver built-in service of the Intera-SDK is used to derive the final joint positions.

The trajectory generation is performed using the joint mode of the same integrated software, which generates a list of waypoints towards that final pose. This happens in the joint space therefore, a joint interpolation is performed.

Finally, since the task to perform involves interacting with the object of interest, the magnet to be used has to attach to the top of the brick. In practice, the surface of the magnet has to remain as flat as possible with the brick surface to ensure better magnetic gripping force. An impedance control scheme will allow for a misalignment of the two surfaces to be corrected by giving a commanded position of height that slightly intrudes into the model bricks. This scheme is modelled as follows:

$$F = K(x_{des} - x_{act}) + D(\dot{x}_{des} - \dot{x}_{act}),$$

from which the desired interaction forces are computed from the desired and actual endpoint velocities and poses. The joint torques are expressed as:

$$\tau = J(q_{act})^T F + \tau_{nullspace}$$

.
The Jacobian and torques are calculated with the built-in software, but the damping and elastic coefficients are custom set.

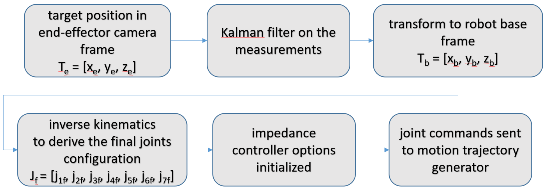


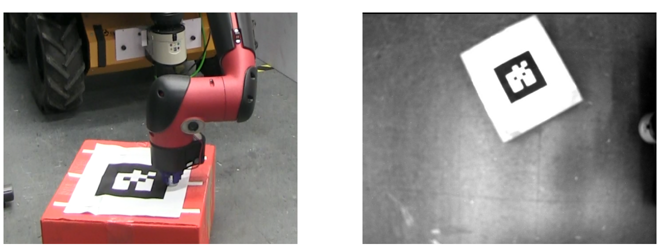
Fig. 2: Process layout

IV. EXPERIMENTS

The testing and validation procedure was based upon these principles and the tests included the following. The first part involved comparing the three detection packages and each camera system to see which ones provide the more robust and precise center of top brick surface position. The second part was to test the smoothness of the trajectory, the inverse kinematics solver adequacy and the impedance system necessity.

For the first one, the system used the ZED camera close to the base of the arm. After several tries the AR code proved to be the ideal for detection and tracking of the center of brick under multiple orientations. The find object 2d was not robust in changes of orientation and the visp auto tracker suffered from the fact that the QR code had to be close and almost parallel to the camera frame. For these reasons the ar track alvar package was chosen.

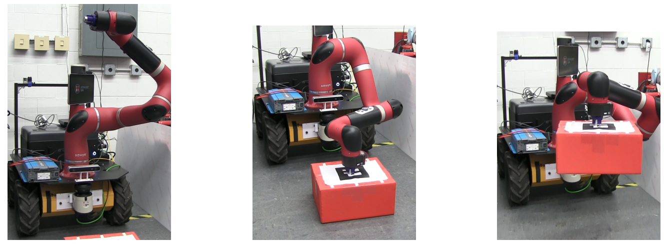
Another comparison critical for the robustness and precision had to be done between the two camera systems. Indeed, using the AR code it was found that the camera on the rover was failing to provide very precise position estimation of the centre when the angle of brick and center of rover was beyond some limit, around 30°, as in figure 3a . On the contrary, the manipulator camera remaining always above the brick, as in figure 3b and having its plane parallel to the top surface, always gave a very accurate estimation.



(a) ZED innacurate detection (b) Manipulator camera view

Fig. 3: The camera system used and manipulator orientation before picking operation

As far as the control of the arm is concerned, the trajectory planner and inverse kinematics performed adequately well and the impedance method proved critically important for the success of the pickup. All the parameters including the stiffness, damping, camera system calibration and tuning, were done and the resulting tuning proved to be precise for various brick positions within reach of the robotic manipulator. The pickup operation was performed vertically after grabbing the brick.



(a) Detection (b) Grabbing (c) Lifting

Fig. 4: The three stage operation of picking the brick

V. CONCLUSION AND FUTURE WORK

In conclusion, autonomous picking of bricks has been proven to require an accurate system of detection, capable of estimating the grabbing position under various orientations. Using a detection code for the purpose is the first step towards the integration of a detection system that utilizes only the shape and color of the brick to find the center. An approach like this, given new competition specifications, will be the next step to implement in the system for the camera on the robotic arm system. The rest of the system parameters adequately perform for the given task. Further modifications will be made to the grabbing magnet to ensure larger touching surface and tuning of the speed of the system to ensure it matches the challenge time given. Another step to be done, concerns placing the stack of bricks on the wall structure which is topic to be dealt with at a later stage.

REFERENCES

[1] "Rethink robotics." https://www.rethinkrobotics.com/, 2018.
[2] "Universal robotics." https://www.universal-robots.com/, 2018.
[3] "Construction robotics." https://www.universal-robots.com/, 2018.
[4] "International federation of robotics." https://ifr.org/ifr-press-releases/news/robots-double-worldwide-by-2020, 2018.
[5] "Mbzirc." https://www.mbzirc.com/challenge/2020, 2018.
[6] H. Ardiny, S. Witwicki, and F. Mondada, "Construction automation with autonomous mobile robots: A review," in *2015 3rd RSI International Conference on Robotics and Mechatronics (ICROM)*, pp. 418–424, Oct 2015.
[7] V. Helm, S. Ercan, F. Gramazio, and M. Kohler, "Mobile robotic fabrication on construction sites: Dimrob," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4335–4341, Oct 2012.
[8] K. Petersen, R. Nagpal, and J. Werfel, "Termes: An autonomous robotic system for three-dimensional collective construction," 06 2011.
[9] S. Magnenat, R. Philippsen, and F. Mondada, "Autonomous construction using scarce resources in unknown environments," *Autonomous Robots*, vol. 33, pp. 467–485, Nov 2012.
[10] J. Willmann, F. Augugliaro, T. Cadalbert, R. D'Andrea, F. Gramazio, and M. Kohler, "Aerial robotic construction towards a new field of architectural research," *International Journal of Architectural Computing*, vol. 10, no. 3, pp. 439–459, 2012.
[11] "find object 2d." http://wiki.ros.org/find_object_2d, 2018.
[12] "Qr code tracker." http://wiki.ros.org/visp_auto_tracker, 2018.

[13] "Ar code tracker." http://wiki.ros.org/ar_track_alvar, 2018.