22nd International Conference
ENGINEERING MECHANICS 2016
Svratka, Czech Republic, 9 – 12 May 2016

INDSUSTRIAL ROBOT VISON SYSTEM FOR MOVING SPHERICAL ELEMENTS

J. Zwierzchowski*

Abstract: *In the application of industrial robots main and still increasing role play automation systems controlling their work. In order to enable the automation and a sense of intelligence we used vision systems. This paper describes industrial robot control software running on a PC. Information about the robot workspace is collected on by a video camera. The main robot task is to search items lying on the special table, and then move them to the place indicated by the operator. We developed appropriate objects searching algorithms for industrial robot. Those algorithms are based on object features (shape and color). We also describe communication between the robot, the computer and the camera. The proposed algorithms are implemented in C ++ (PC) and RAPID (robot) languages. The paper presents examples of using the application and tests system performance. It should also be noted that developed and described in the article system is based on very cheap to buy parts.*

Keywords: industrial robot, vision system, recognition, path generation

1. Introduction

The main task of the industrial robot used in this application is to move round parts, from an undefined position to a precise position determined by the system operator. This means that the place from which the element is taken must be indicated by software (Kovar, J et al., 2013), so the system will analyze the robot workspace and generate commands of its movements.

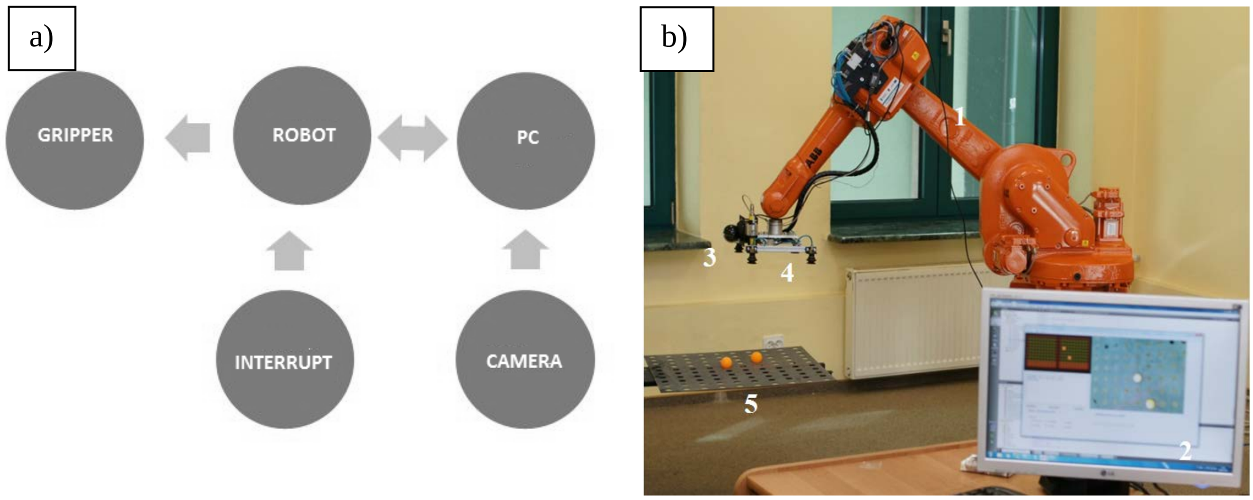

Fig. 1: a) System architecture for moving spherical elements; b) View of research station to move balls: (1) manipulator ABB IRB 1600, (2) managing application interface, (3) digital camera, (4) specially designed vacuum gripper, (5) work table with balls

The system architecture is shown in Figure 1a. Central point of this architecture is an industrial robot, which also is a communication server between robot and the PC application. In addition, the robot uses

* Dr inż. Jarosław Zwierzchowski, PhD.: Faculty of Mechatronics and Mechanical Engineering, Kielce University of Technology, Al. Tysiąclecia Państwa Polskiego 7; 25 – 314 Kielce; PL, j.zwierzchowski@tu.kielce.pl

digital input and output signals to control a vacuum gripper. There is also a digital camera, that works directly with the software installed on PC. Example of experimental station is shown in Figure1b.

Description of the system can be ordered and presented in several steps. The first step is to run the system. Then, after turning on, the robot sets its TCP (Tool Center Point) at the position, which is named *CameraHome*. This is one of the few points of the robot manipulator coordinates, which are permanently stored in the robot controller. This point is obtained in calibration process of the system. Locating at this point robot gripper ensures that captured image from the digital camera shows whole worktable, and *Z* coordinate of manipulator TCP is in the proper distance from one to table (Fig. 1b). This gives the possibility of calculating a correct diameters of spherical shapes, which ware found. At this point, the application communicates with the robot controller. The next step is to analyze the workspace. The application program searches appropriate elements using features as follows: color, shape and size of elements. These parameters are entered to the system by user. After identifying and recognizing the round objects, the control application takes action by sending the appropriate command to the industrial robot and then waits for the command performing movement. This order is appeared by system operator. The next step of the system is executing the robot program. After all activities related to transfer elements, the robot returns to the position *CameraHome* and again waits for orders coming from the operator.

2. Functional schema of the system and robot communication

In order to test system correctness of detecting and determining spherical objects position coordinates we designed and made a robot worktable. As can be seen in the drawing (Figure 2a) the worktable has holes, which are made in specific places (in the figure marked only selected holes). This approach greatly simplifies determining the position of an object in the shape of ball, inserted into the hole on the worktable. The figure (see Figure 2a) also shows the place of attached the Cartesian coordinate system $X_S Y_S Z_S$ onto worktable.

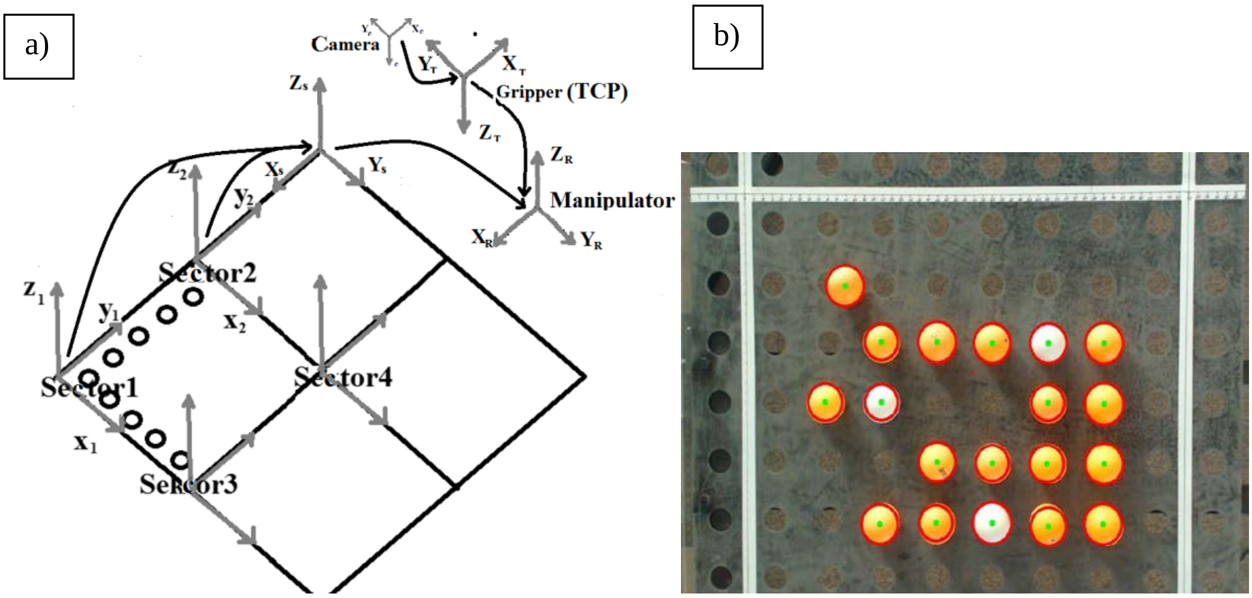

Fig. 2: a) Distribution of sectors onto the tablework. Gray marked the coordinate systems of each sector: worktable $X_S Y_S Z_S$, manipulator $X_R Y_R Z_R$, gripper $X_T Y_T Z_T$, camera $X_C Y_C Z_C$, b) The result of the Hough algorithm (a fragment of the worktable); (green) centers of detected objects, (red) detected round shapes

In described system, it is assumed that the gripper can "take away" spherical element only from the designated places. It means that there are cut grid holes in the table (see Fig. 2a). For simplicity, it is assumed that the table has N holes along both X_S and Y_S axes. Denote

$$p(pozX, pozY), \qquad (1)$$

where:

$$pozX = 0,1,2 \ldots N-1, \; pozY = 0,1,2 \ldots N-1 \qquad (2)$$

the essential points of the grid of holes. These points can be interpreted as the specific number from holes in the table. The actual position p_s in regard to coordinate system $X_sY_sZ_s$ (worktable) we can obtain form equation

$$p_s(x,y) = p[(pozX+1)\cdot f, (pozY+1)\cdot f], \tag{3}$$

where f is constant distance value between the centers of the holes in the worktable, expressed in mm.

The table is located in the working space of the manipulator in this manner, that it is impossible to set robot in a singular configuration. The robot is equipped with vacuum gripper. Initially, the gripper had only one sucker directly attached to the manipulator wrist. This approach meant that the manipulator had small movement range and a fragment of the table was located outside its working area. In order to eliminate this drawback, we designed and built a new gripper, increasing the manipulator movement range. Currently, the gripper is made from aluminum profiles arranged in the shape of a rectangle. At each of the rectangle corners there is mounted sucker. Robot controller controls each of the sucker separately.

The worktable is divided into four sectors and each sector has 25 holes (5 x 5 grid). The origin of each sector is called base position. Those base positions are permanently stored in the structure of the robot program. To each sector is assigned exact sucker.

Having the scene describe, now we should think which sucker would be move and where. Now new real points coordinates p_b^i are calculated related to coordinate system of sector base position i, where $i = 1,2,3,4$, and i is obtained from algorithm as follows:

$$i = \begin{cases} 1, & pozX \bmod b + pozY \bmod b = 0 \\ 2, & pozX \bmod b > pozY \bmod b \\ 3, & pozX \bmod b < pozY \bmod b \\ 4, & pozX \bmod b + pozY \bmod b = 2 \end{cases} . \tag{4}$$

The value i also indicate the number of sucker, which system should use during object movement. Real points coordinates related to sector i coordinate system can be calculated from equation

$$p_b^i(x,y) = p\left[(d_x+1)\cdot f, \left(d_y+1\right)\cdot f\right], \tag{5}$$

where

$$d_{x,y} = \frac{pozX, pozY}{b} \tag{6}$$

and f is constant distance between the centres of the holes in the worktable, d_x, d_y are hole coordinates (numbers) refer to i base coordinates system, $b = \frac{N}{2}$ and $b, d_{x,y} \in \mathrm{C}^+$.

3. Implemented vision system

For the purpose of the system was developed and implemented a simple algorithm to recognize objects, which projection on the worktable has a circular shape. The following figure (Fig. 3) shows subsiquent steps in the proposed image processing system. At the beginning propose system recognizes color. User system can indicate, which color is important to move and which color the system should avoid. After that, recognition algorithm changes original capture image into grayscale image. The next very important step is to perform edge detection procedure. For this task we use Canny transform (Canny, J., 1986). Then it is performed Hough (Pedersen, S. & Just, K., 2007) transform for circles. After this operation we have the list of circles center points and its radii obtained. Those points are transform form image coordinates system to manipulator coordinates system. Having all information about detected round objects position the system can generate moving instruction and send them to robot controller.

Figure 2 shows fragment of the worktable with detected table tennis balls. The manipulator transfers only orange balls. In figure 3 there are shown all used sequences in image processing task.

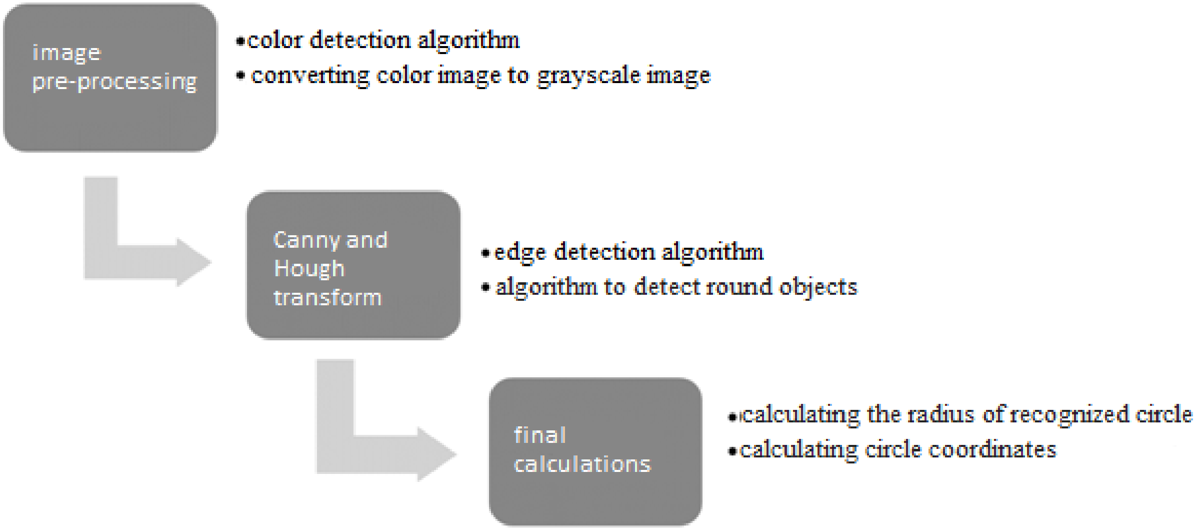

Fig. 3: Next steps in image processing used in designed system application.

4. Conclusions

We conducted tests using industrial robot ABB symbol IRB1600 with six degrees of freedom. The robot was equipped with a vacuum gripper and a digital camera Logitech symbol 920C. The camera sent the image to the computer at a resolution of 1920 x1080 pixels. In manipulator workspace was mounted worktable, which was a plate with cut-out holes spaced 60 mm (the centers of these holes), each hole was 18 mm in diameter. These 100 holes ware arranged in 10 by 10 grid. The tests had been prepared to use table tennis balls, each ball was 40 mm in diameter. We did 20 trials series. Every attempt was made at 12 the same set of balls located on the worktable. In each series there were orange balls (10 pieces), which robot had to move and white balls (2 pieces), which robot had to ignore.

The system software correctly rejected the white ball. Total recognition rate calculated as the average value of all the series, was 0.783. We can conclude that the system works properly, but in the future in order to raise this coefficient value we will install illuminator and add filtering algorithm (Janecki et al., 2015). We will test this system with delta robot (Łaski et al., 2015).

We managed to integrate an industrial robot with the application running on a PC, using the protocol TCP/IP. All programs were written in the RAPID ABB robot and C++ languages with OpenCV and Qt library. We would like also mention, that described system is primarily focused on testing communication reliability of all components and the correctness of the manipulator static paths generation.

Described system is an attempt to show program, which can be used for example in sorting or palletizing industry applications.

Creating system can be regarded as a testing platform for further research on machine vision in robotics. There can be also testing new algorithms like image analysis, communication, scheduling, artificial intelligence.

References

Canny, J. (1986) A computational approach to edge detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on: pp. 679-698

Janecki, D. & Zwierzchowski, J. & Cedro, L. (2015) Separation of non-periodic and periodic 2D profile features using B-Spline functions. Metrology and Measurement Systems t: 22, issue: 2, pp: 289-302

Łaski, P. & Takosoglu, J. & Błasiak, S. (2015) Design of a 3-DOF tripod electro-pneumatic parallel manipulator. Robotics and Autonomous System issue: 72, pp: 59-70

Pedersen, S. & Just, K. (2007) Circular Hough transform. Aalborg University, Vision, Graphics, and Interactive Systems.

Kovar, J.; Andrs, O.; Dalecky, F. (2013) Using of Hough Transform for Automatic Robot Sorting System 19th International Conference on Soft Computing MENDEL 2013 pp: 375-378