

The maintenance, commissioning of FANUC M-1iA 0.5A type selecting robot and implementation of iRVision picture-recognizing procedure

Gyula Korsoveczki^{1,*}, Dr. habil. Géza Husi², and Timotei István Erdei³

¹Mechatronics Department, Engineering Faculty, University of Debrecen, Hungary

²Mechatronics Department, Engineering Faculty, University of Debrecen, Hungary

³Mechatronics Department, Engineering Faculty, University of Debrecen, Hungary

Abstract. The topic of the given project is based on the maintenance, commissioning and implementation of iRVision picture-recognizing procedure, and the operation of the associated program of the FANUC M-1iA 0.5A type delta tripod designed selecting robot that can be found at the Mechatronics Department, Engineering Faculty of the University of Debrecen. Keywords — FANUC; M-1iA 0.5A; delta tripod; robot; iRVision; picture-recognizing

1 Introduction

The permeation of robots is more considerable in every area of industry. With the development of the technic and growing of demand there is need for automated production lines. The velocity and precision of these lines gives industrial robots. With the opportunity of this project I could work with that kind of industrial robot I have always wanted to. That is why I chose the mentioned FANUC M-1iA 0.5A type delta tripod designed selection robot that can be found at the Mechatronics Department, Engineering Faculty of the University of Debrecen. After I had completed the maintenance operations, I measured the tool used by the robot. Next, I optimizationed the sphere of the workplace and I measured the user frame that I used during the project. After this operation, I designed several tools and elements for the robot. These metal elements make the work desk stronger and ensure a higher level of stability for the construction that supports the work desk. The subjoined CD contains the complete plans. Furthermore, I wanted to know the working of the iRVision system of FANUC and the implementation of picture-recognizing procedure that I wanted to show in my task. As the result of these operations, the sphere of work of the robot was broadened and the functioning became more efficient and competent in educational purposes.

2 Description of the system

2.1. FANUC M-1iA 0.5A type robot

The robot I used during my task is a delta tripod designed robot, that possesses 6 degrees of freedom. This model is versatile and nimble, and it can achieve excellent work rates on complex component assembly

operations. The robot is capable of working at any angle, so it is good choice for those applications that demand high level of repeatability and flexibility.

The scope of the application is electronics, food products industry, manufacture of medical instruments optic and medical industry and precision mechanics [1].



Fig. 1. FANUC M-1iA 0.5A [1]

The technical specifications of the robot:

- Number of controlled axes: 6
- Maximal reach: 280 mm
- Repeatability: +/- 0,02 mm
- Mechanical weight: 23 kg
- Maximal load capacity at wrist: 0,5 kg
- Maximal velocity: 4000 mm/s [1]

2.2. FANUC R-30iB Mate controller

The robot is controlled by FANUC R-30iB Mate Open-Air type of controllers. It is compact and easy to integrate into single robot production cells. Unlike Windows-based controllers, the R-30iB types use

* Corresponding author: korsoveczki.gyula@gmail.com

FANUC's own propriety software. This allows that the system is more stable and it is protected against the most cyber virus and attacks of hackers. Furthermore, its controller advantage is that there is no need frame system to run [2].

The controller also suits the requirements of industrial environment, because it has suitable protection against wetness and dust. To use the iRVision system easily, an Ethernet port can be found on the controller. The main parts are a switch with two position (ON-OFF), an emergency toggle with red colour and an operation mode changer green toggle that can be actuated by a small key. This green toggle is the responsible for the running of the cycle program [2].



Fig. 2. FANUC R-30iB [3]

2.3. FANUC iPendant Touch

The robot can be programmed in two ways. One of these ways is the Karel program language. To use this one, a PC is needed. The other opportunity is the iPendant Touch hand controller that is powered by FANUC. As this hand controller uses block style program language, there are more options to program the robot and it is faster than in case of Karel language.

The iPendant Touch has ergonomic shape, touch screen and its programming environment is user-friendly. To further enlargements it contains function toggles and there can be opened two windows on the big colourful screen at the same time. The hand controller has a 4D designer option and USB input to connect external store or mouse [4]. The main parts are the big colourful screen, the keyboard, the safety Deadman switch with three positions, an emergency toggle and a switch with two positions that can enable the iPendant Touch.



Fig. 3. FANUC iPendant Touch [Own figure]

2.4. SONY XC-56 type camera

The system includes an industrial monochromatic camera that is powered by SONY. This camera is compatible with the iRVision picture-recognizing system. It has the duty to accomplish the visual part of the implemented procedure.

The main technical specifications of the camera:

- High resolution: VGA resolution (648 x 494)
- High rate scanning: 120 frames / second
- Minimum illumination: 0,5 lux
- High shock and vibration resistance [5]



Fig. 4. SONY XC-56 camera [5]

3 The maintenance operations

3.1. Reset of the serial pulsecoder

During the task, my first problem was related to the batteries of the serial pulsecoder. The robot has 6 moveable axes. These axes are moved by 6 AC servo motors. To give their actual position, there are incremental pulsecoders in the robot. These kinds of pulsecoders need stationary electric supply. Without this, the result of the counting perishes [6]. That is why the robot includes 4 Industrial ALCALYN batteries. However, these batteries have gone flat since the last use of the robot. In this case a special alarm turns up, namely SRVO-062 BZAL alarm (Group: i Axis: j). It means Battery Zero Alarm. To solve this problem, changing the batteries is not enough, because there is need to delete and unlock this one in the system. With using of the iPendant Touch, we need to enter into the menu of the robot with pushing of the MENU button. After that, there is need to select NEXT (0) and the SYSTEM (6) options. Finally, we can reach the internal library of the robot with pushing of the VARIABLES (2) button. With the ITEM function we can search according to serial numbers. In this case, the number of the searched item is 308. This is \$MRC folder. In this folder we need to find the \$SPC_RESET variable with the serial number of 14. It means SERIAL PULSECODER RESET. When it is found, we have to change the value FALSE to value TRUE. Completing this operation, there is need to push the Deadman switch into middle position and the RESET button at the same time. As the next step, there is need to shut down the robot and disconnect to the electric circuit. When we turn on the robot, the axes will be moveable and we can use the JOINT coordinate-system [7].

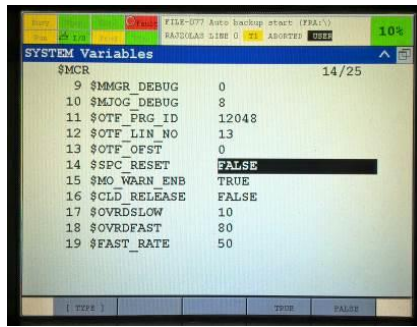


Fig. 5. The \$SPC_RESET [Own figure]

3.2. The MASTERING and the CALIBRATION

Following the reset of the serial pulsecoder, it was necessary to perform the MASTERING and the calibration procedures. The actual position of the robot depends on the values of the pulsecoders. The corresponding count value is obtained from the rotational speed of the pulsecoder connected to the relevant motor and the rotation angle within one rotation. By right of this principle we can give values to the pulsecoders with the MASTERING operation.

This procedure is accomplished at the factories, users do not need to perform at every use. However, it is necessary after motor, pulsecoder and cable replacing, moreover, in case of battery problems [8].

During my task I accomplished the QUICK MASTERING operation on the robot. This procedure is implemented in a position that is defined by the user. QUICK MASTERING uses the fact that the absolute value of a rotation angle within one rotation will not be lost. This kind of MASTERING can be used, if the pulsecoder count value is lost, for example a low voltage detects on the backup battery for the pulse counter. However QUICK MASTERING cannot be used, after the serial pulsecoder is replaced or the mastering data is lost from the robot controller [8].

To accomplish QUICK MASTERING, there is need to select the SYSTEM option and then we have to choose the MASTER/CAL opportunity with pushing the button 3. Here we can reach all of the MASTERING options. First of all, it is necessary to move the robot axes into a random position. To complete this operation, we need to select the JOINT coordinate-system with the COORD button. The next step is to push the RESET button and the Deadman switch into the middle position at the same time. With this operation, we can unlock the FAULT alarm.

To move the robot axes, it is needed to push the Deadman switch, the SHIFT button and one of the direction buttons at the same time. These direction buttons can be found on the right side of the iPendant Touch. In this case, the TOOL CENTER POINT (TCP) moves into the required direction. The direction buttons are the following:

- +X; -X; +Y; -Y; +Z; -Z: Moving along the axes of the Descartes coordinate-system.
- +X °; -X °; +Y °; -Y °; +Z °; -Z °: Rotating around the axes of the Descartes coordinate-system.

To select the QUICK MASTERING, we need to use the navigation buttons and to complete it, we have to press ENTER.

After performing this operation, pulsecoders get count values. However, it is not enough to give values to the pulsecoders, because we have to save these values. There is the CALIBRATING option for this purpose. To perform this operation, it is important to choose the CALIBRATE opportunity [8].

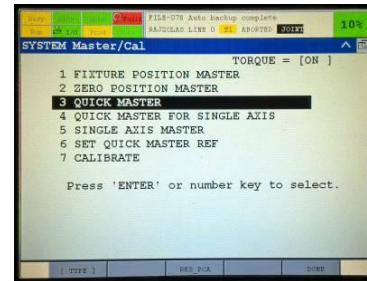


Fig. 6. The MASTER/CAL menu [Own figure]

During my task I completed all of the described operations. As a result, I achieved the aim that the robot can move in WORLD coordinate-system. To define and present the directions, I created an illustration that contains these directions. The illustration can be found on the construction that supports the robot. Knowing of the directions enables faster and easier robot programming for the users.

4 The commissioning of the robot

4.1. Measuring of the tool frame

Finishing with the maintenance, I had the opportunity to commission the robot. The first step of commissioning was the measuring of the tool frame.

To work the robot properly, there is necessary that we perform this operation. With measuring of the tool frame, we teach the robot the TOOL CENTER POINT. Knowledge of this information is essential for the robot because the robot gets information about the geometric parameters of the tool, which is found on it.

To accomplished the measuring, it is necessary to enter into the MENU of the iPendant Touch. After that, we need to select the SETUP (6) option and within that the FRAMES possibility with the button 3. Here there is need to open the [OTHER] function to reach the TOOL FRAME platform. Following this procedure, the system gives us a list about the previously measured tools. We have the opportunity to modify these tools but we can also define new tools. To measure a new tool, we have to move the cursor on an empty place and then with the CREATE order we are start the operation. Opening of the [METHOD] function helps us to choose the method of the tool measuring. The system offers to us 4 possibilities. These are the Three Point, Six Point, Two Point + Z and the Direct Entry.

While I was dealing with my task, I used a 3D printed tool and I measured it with the Three Point

method. The essence is of this method, that we touch with the target tool the chosen reference point from three different directions and with three different orientations. To complete the procedure, there is need to define 3 points which are APPROACH POINT 1, 2 and 3. These points are the mentioned touching points. We have to move the TCP into the demanded points and then it is necessary to save them. We can do this with pushing the SHIFT button and selecting the RECORD order at the same time. We have the possibility to comment the measured tool into the COMMENT space.

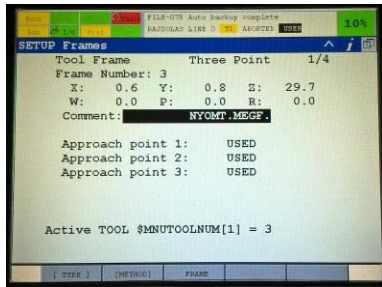


Fig. 7. Measuring of the tool frame [Own figure]

4.2. Optimization of the work sphere

The next purpose of my task was the optimization of the work sphere. Thanks to this operation, the scope of the work of the robot broadened and with the setting of the work sphere the precision has increased.

Before the procedure, the robot had possessed a work sphere but from the point of the view of my project it had been unsuited.

The first problem I discovered that the work table was not horizontal, because it possessed a thereabouts 20 degrees. This problem made the robot unsuited for the task solution because I wanted to implement a 2D viewed picture-recognizing procedure. To solve this problem, I created a construction made of performed aluminium. It was compatible with the supporting construction of the robot. To fix the fittings, I used small metal components that have screw-threads. With constraining or slacking the components the work table proved moveable along the splints.

As the work table I used a metal practise table powered by KUKA GmbH. It is stable, it includes height differences and there are practise figures on the surface. Another problem I discovered that the maximal reach of the robot is 280 mm, so it was not able to reach the surface of the work table. To solve this problem, I have lifted up the new work table. Being stable and adjustable are very important criterions against the work table. I accomplished these viewpoints with 4 shafts that possess screw threads. Their diameters are M6 and they are 120 mm long. The screws are compatible with the small metal components and with the whole construction. They warrant X and Y direction mobility by the WORLD coordinate-system. To fix the practise table, I used screw-nuts and vapid spacers. Thanks to that, the position of the practise table is variable vertically. It means that the Z position is adjustable.

As a result of the performed changeovers, I could set the work sphere of the robot optimal. The directions of the mobility are presented in figure 8.

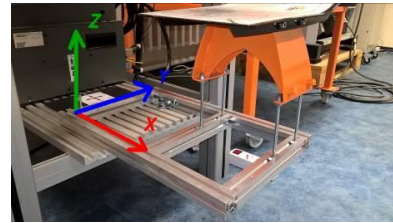


Fig. 8. The new work sphere [Own figure]

4.3. Measuring of the user frame

After the suitable constructing of the work sphere, I had the opportunity to measure the user frame that was used by the robot. In addition, I laid down the used coordinate-system and I defined the directions of this.

The measuring of the user frame can be reached from the menu of the iPendant Touch with the SETUP (6) option. To get this, there is need to open the FRAME (3) opportunity and then it is needed to select the USER FRAME possibility with the opening of the [OTHER] function. Here we can modify before measured frames or we can measure a new one. The [METHOD] function offers the Three Point, Four Point, and the Direct Entry operations.

During the solving of the task I performed the Three Point method. In this case we need to define the origo of the user frame and two other points which are in X and Y directions. These directions are the same with the directions of the WORLD coordinate-system.

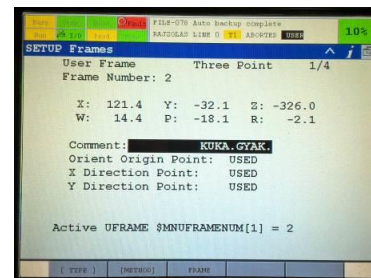


Fig. 9. Measuring of the user frame [Own figure]

5 Creating of the iRVision procedure

The iRVision is the FANUC's own PLUG & PLAY visual sensor system. The robot possesses a camera and a picture-analysing process that is connected to the camera. These items provide vision for the robot. It is completely integrated into the R-30iB Mate controller. As a result of this, the iRVision can be used easily and it offers a high level of flexibility to the users. Thanks to the iRVision system, the robot is suitable for selecting, complex component assembly operations and it is able to recognize things with optional scale, colour and position in 2D and 3D as well [9].

Meanwhile I was dealing with my project, I wanted to implement a picture-analysing procedure that

recognizes a given 2D coordinate-system with the use of the camera. After that, it touches the center point of the coordinate-system and it draws around the whole triangle which is placed in this one. However, this triangle is invisible for the robot. To accomplish these operations, we need a robot program that connects the iRVision system to the robot.

The speciality of the iRVision is that it has a direct connection to the controller. In case of computer connection, there is need an Ethernet cable and a PC that can run Microsoft Internet Explorer.

The first step of the implementation is connecting the PC to the controller of the robot with Ethernet cable. For the communication, we need to overwrite the IPv4 address of the PC to 192.168.0.11. Following this operation, it is necessary to open the Internet Explorer and we can reach the main software of the controller with the address of 192.168.0.1. There are special components for the Internet Explorer which are essential to work the system. These components can be downloaded with clicking to the iRVision Setup menu.

From the appearing menu we should choose the CAMERA SETUP TOOLS opportunity with the help of [TYPE] option. After that, with using of the CREATE command we can create a new one. Here we can define the settings of the used camera. In addition, we have to select fix camera option and set the exposition time.

The next step is to complete the calibration of the camera with the CAMERA CALIBRATION TOOLS function. To do this, we need the FANUC's own calibration grids. Without this, we cannot accomplish the operation. The main point of the calibration is that the robot optimizes the settings of the camera itself to get the maximal efficiency. Since I didn't possess with calibration grids, I used the basic settings of the camera.

After all these operations, there is need to create a new picture-recognizing procedure, namely VISION PROCESS TOOLS and within this a 2-D SINGLE – VIEW VISION PROCESS. The name of the process I created is "KRESHAROMSZOG".

In this opportunity, we have to define the camera calibration, the exposition time and the criterions of the finding of the searched components. Furthermore, there is need to select the previously measured user frame and we have to define a reference position.

The menu contains a GPM LOCATOR TOOLS option. With using of it, we can teach various objects to the robot. To teach them, we have to take a picture with the SNAP command. Following this operation there is need to trace the object that we want to teach to the robot. In addition, we have the possibility to mask the parts which are unimportant from the point of view if searching, as I did in case of the triangle. This operation can be completed with the TRAINING MASK command.

Important tools are the CONTRAST TRESHOLD, ORIENTATION AND SCALE OPTIONS. With the first one, we can search by colours. In case of the second one, we have the opportunity to find objects in various orientations. The third one enables us to search objects by their scales.

With the end and saving of the edition comes into existence the picture-recognizing procedure that can be testing straight. Meanwhile the testing, the software shows visually the progress and the result of the searching.

With completing of the presented steps, my procedure could be accomplished successfully. I taught the coordinate-system to the robot as a searchable object.

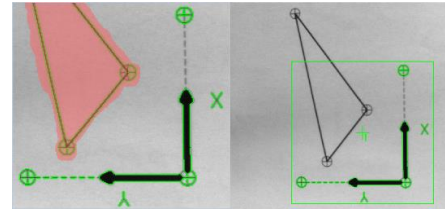


Fig. 10. The searched and found object [Own figure]

6 The associated robot program

6.1. Description of the program code

As the last step of my task, I wanted to create an associated robot program to the picture-recognizing procedure. Thanks to that, the robot can use the mentioned procedure in practice. With running the program, the robot is able to recognize the taught coordinate-system which is can be found on the work table of the robot in any position but in given orientation. Furthermore, the robot can touch the origo of the coordinate-system and after that it can draw around the triangle in the coordinate-system. In this case the triangle is invisible for the robot, so the position of the triangle depends on the found coordinate-system.

The programming happened with the using of iPendant Touch. It is much more suitable and faster than the usage of Karel program language.

In pursuance of this as the first step, I defined the previously measured tool frame and user frame in line 1 and 2.

As the third step, I implemented a view point for the procedure. In this point the TCP do not disturb the vision of the camera. The type of the motion is linear (L), the velocity is 100 m/s and at the end of the motion, the robot stops (FINE).

In line 4, I connected the picture-recognizing procedure to the robot program with the VISION RUN_FIND command. The purpose of the searching, I gave the name of the visual process which is (KRESHAROMSZOG). The camera of the robot checks the work table according to the picture-recognizing procedure to find a taught object.

In case of successful detecting, the robot saves the coordinates of the object into an OFFSET file in line 5. However, if it does not find anything, it creates a jump (JMP) onto a marked label (LBL [1]) to skip over the part of the program code which belongs to the finding.

With the commands which can be found in lines 6-8, the robot touches the origo of the coordinate-system. After that, with the commands in lines 9-14, the robot

draws around the triangle according to P [4], P [5], P [6], P [7], P [5] and P [4] points.

Finally the LBL [1] can be found in line 15, and with line 16 and the [END] command the robot returns into the P [1 : VIEW] point and finishes the program. The program code is the next:

```

1 : UFRAME_NUM=2
2 : UTOOL_NUM=3
3 : L P [1 : VIEW] 100mm/sec FINE
4 : VISION RUN_FIND 'KRESHAROMSZOG'
5 : VISION GET_OFFSET
   : 'KRESHAROMSZOG' VR [1] JMP LBL [1]
6 : L P [2 : APRCH] 80mm/sec FINE
   : VOFFSET , VR [1]
7 : L P [3 : PICK] 80mm/sec FINE
   : VOFFSET , VR [1]
8 : L P [2 : APRCH] 80mm/sec FINE
   : VOFFSET , VR [1]
9 : L P [4] 100mm/sec FINE
   : VOFFSET , VR [1]
10 : L P [5] 100mm/sec FINE
    : VOFFSET , VR [1]
11 : L P [6] 100mm/sec FINE
    : VOFFSET , VR [1]
12 : L P [7] 100mm/sec FINE
    : VOFFSET , VR [1]
13 : L P [5] 100mm/sec FINE
    : VOFFSET , VR [1]
14 : L P [4] 100mm/sec FINE
    : VOFFSET , VR [1]
15 : LBL [1]
16 : L P [1 : VIEW] 100mm/sec FINE
[END]

```

6.2. The flowchart diagram

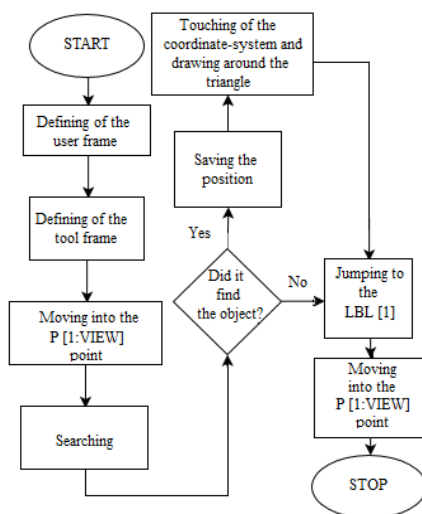


Fig. 11. The flowchart diagram [Own figure]

7 Conclusion

The purpose of my task was to accomplish the maintenance and the practice oriented commissioning of the robot. Furthermore, I wanted to test and submit the iRVision system of the robot with the help of the

associated program. As a result, I got a system that works well and I accomplished the purposes. There are other possibilities, for example creating various target specific tools or implementation of external light source and conveyor line. Moreover, the recalibration of the camera is more important to make the precision and efficiency better and to improve the scope of the work. However, the software part can be broadened without any problem with new part of program codes and functions.

Acknowledgement

I want to render thanks to Dr. habil. Géza Husi and to Timotei István Erdei in order to be consultant authors, and to submit willingly the necessary devices. The publication was supported by the project EFOP-3.6.1-16-2016-00022. This project was co-funded by the European Union, from the European Social Fund.

References

1. FANUC Hungary Kft., „M-1iA/0.5A,” [Online]. Available: <http://www.fanuc.eu/hu/hu/robotok/robotsz%c5%b1r%c5%91-lap/m1-sorozat/m-1ia-05a>. [Accessed: 27.11.2017.].
2. FANUC Hungary Kft., „R-30iB controller.” [Online]. Available: <http://www.fanuc.eu/hu/hu/robotok/tartoz%c3%a9kok/r-30ib-vez%c3%a9rl%c5%91>. [Accessed: 27.11.2017.].
3. RobotWorx, „FANUC R-30iB Open Air Cabinet,” [Online]. Available: <https://www.robots.com/fanuc/controller/r-30ib-open-air-cabinet>. [Accessed: 27.11.2017.].
4. FANUC Hungary Kft., „iPendant Touch,” [Online]. Available: <http://www.fanuc.eu/hu/hu/robotok/tartoz%c3%a9kok/r-30ib-vez%c3%a9rl%c5%91/ipendant-touch>. [Accessed: 27.11.2017.].
5. SONY, „XC56,” [Online]. Available: <https://pro.sony.com/bbsc/ssr/product-XC56/>. [Accessed: 27.11.2017.].
6. Dr. P. Boza, Dr. J. Pintér, *Manufacturing automatization “Gyártásautomatizálás,”* Győr, 2011.
7. FANUC Robotics America, Inc., „M-1iA Operator Manual,” [Online]. Available: <https://www.scribd.com/document/350205121/M-1iA-Operator-Manual-B-83084EN-08>. [Accessed: 27.11.2017.].
8. FANUC Robotics America, Inc., „R-30iA Mate CONTROLLER,” [Online]. Available: http://www.msamc.org/aimss/documentation/pdf/manuals/lr_mate_manuals/R30iA%20Mate%20Controller.pdf. [Accessed: 27.11.2017.].
9. FANUC Hungary Kft., „Visual sensor system,” [Online]. Available: <http://www.fanuc.eu/hu/hu/robotok/tartoz%c3%a9kok/1%c3%a1t%c3%a1s>. [Accessed: 27.11.2017.].