Journal of Engineering and Fundamentals
Vol. 2(1), pp. 1-12, June , 2015
Available online at <http://www.tjef.net>
ISSN: 2149-0325
<http://dx.doi.org/10.17530/jef.15.03.2.1>

Improving Object Classification and Application Software with Mitsubishi Robotic Arm

Fatih ARSLAN *

Department of Electrical and Electronic Engineering Graduate Student, KSU, Turkey
fatih@arerotomasyon.com

Mahit GÜNEŞ

Department of Electrical and Electronic Engineering, KSU, Turkey

Article history	<p>Robotic arms are widely used in flexible manufacturing systems and especially in such environments which threaten the human health. In robotic software it is aimed to do some certain movements as well as moving in changing circumstances during the production process. In this study FESTO and COGNEX cameras are used and a software done on Mitsubishi robotic arm in order to know and classify the objects. As a result of this classification the robotic arm is used integratedly with the camera and the position is controlled sensitively with a %0,5 error. In this study four axis scara robotic arm unit in Mitsubishi Electric laboratory Turkey was used and a robotic arm program which is suitable flexible manufacturing systems was prepared. In this software similarity algorithm of image process was used. Objects were caught according to the similarity algorithm and leaved to a certain place. The camera and the coordinates of the robotic arm positions were calibrated to each other. In this study RT Toolbox software was used.</p>
Received: 28.01.2015	
Received in revised form: 14.02.2015	
Accepted: 16.02.2015	
Key words: Robot Arm, Image processing, image extraction, Pattern recognition.	

* Correspondence: fatih@arerotomasyon.com

1. Introduction

Nowadays robots are used in hundreds of application areas such as installation procedure, welding process, carrying, packing, polishing and trimming, painting and cleaning, oblique process and measurement procedures. There are some orientation software been improvised to design the robotic arms to be appropriate for the flexible manufacturing systems. The robotic software is originated with the help of the orientation software which can do the different tasks at the same time. Robotic arms could also locate the defective products on the production line because they give feedback. So they help the defective products taken out of the line.

F, Şenel [6], Süleyman Demirel University, worked on locating the defective products on the production line with the robotic arm using the embedded systems and image processing. Craig JJ [5] has written a book on robotics and worked on kinematic on robots and inverse kinematics.

With this study we used FESTO and COGNEX cameras on Mitsubishi robotic arms and scara robots, used the image processing algorithms on the cameras, worked on different robotic movement software.

Test 1: Working with the Festo Smart Camera

FESTO camera is used for this study. First of all, the FESTO camera is introduced. To see and change the IP settings of the camera the SBO-Device Manager program is run. In this program cookies for the FESTO camera could be seen. When the camera is scanned in the program, it must be seen as “green” in the list. If the light is “red” the IP settings must be done again.

Image processing with the camera is done using two programs. After introducing the camera Check Opti program and the registered algorithms of image processing are

used and the image is initiated.

The Settings of Camera Configuration in CheckKon Program:

To adjust the settings of the camera configuration, run the Program Files\Festo\CheckKon program. Do the following settings adjustment:

Lighting\ Duration of lighting = Manual
ve 500 ms
I/O Configuration\Internal I/O
Interface\Function at output O2 = External
lighting
I/O Configuration\Ethernet
interface\Web Viewer = On
I/O Configuration\Ethernet
interface\Telnet service\ Telnet Server =
On

In System Parameter enter the parameter settings above. This is very important. For example; which bit will be activated, web access settings, TCP port number and Modbus service settings are all done in this section.[2]

Loading a Program in CheckOpti:

To identify the objects in the camera and use the image processing algorithms on these objects the CheckOpti program is run. When it is on, the camera should be connected. Choose Identify SBO-Q and decide on the active camera.

To be connected and disconnected the camera **Device\Connect or disconnect from device could be used**. Before loading a program to the camera this connection should be done.[2]

Then, choose **View\Check program manager**. In the pop-up menu left part shows the requested program, right part shows where to load it. The requested active program is chosen from the left part and the target program from the right part. The **Copy to device** button is pressed and the program is loaded.

If the process is done reverse, a program could be loaded from the camera to CheckOpti.[2]

Check Opti Program

This program is used to image processing algorithms and identify the objects with FESTO camera.

Fig. 1: Check Opti Object Settings

(1) Displayed pieces' specifications	(4) Coordinate system
(2) Appointed cookies for the piece	(5) Status bar
(3) Image of the piece	

Fig. 2: Check Opti toolbar

- (1) Sends a trigger signal to the camera.
The camera gets a new image.
- (2) Goes to the first piece.
- (3) Goes to the prior piece.
- (4) Goes to the addressed part number.
- (5) Goes to the next part.
- (6) Goes to the last part.
- (7) Part specialties.
- (8) Deletes the chosen part.
- (9) Teach mode on.
- (10) The camera or image mode of the piece is on.
- (11) To zoom.
- (12) Places the piece on the screen.

In this article color perception will be done and for the red products good piece

information will be sent to the robot as the output information of the camera.

Choose Add color dedection to color perception algorithm. Choose the color perception part on the piece. On this section, using the teach part identification could be done. When the perception is done the piece is sent to the good product area. The others are defined as bad products. Define the outputs for the good products on *Function / Set output O1*.

When the Set output O1 is connected to the access of the robot and the color information is settled the robot will start working.

Fig. 3: Piece Perception Good Product – Bad Product

In image processing, the processes are done with the Part contour, Part List, Teach-Data screens. Part contour screen helps the image process algorithms to be applied on a certain piece. In Part screen there are the lists of bad and good pieces. In Teach –Data section the images taken with Part contour are taught on image process methods in details.

If it is wanted, the image information could be sent to a certain data register with the Modbus module. The Flagword address should be written on the yellow part FW tool box part. It must be one of the receivers' addresses. After that, the numerical value will be sent to the receiver's Modbus address.

2. Application Tests

Test 1 Packing Unit

An orientation software is done to be used for a robot in packing unit. With this software the products coming from the line will be packed. But the packing is nonstandard. And the products are nonstandard as well.

There are 3 different types of products on the line. The camera on the line will help to distinguish the products by size sensors. The product information will be sent to 4200 register address via Modbus sending 50-51-52 values. So, the robot program will be able to do pack these 3 different products.

When the 1st type of product comes;

The robot will place the first product. For the second box the product will be turned in 180 degrees.

When the 2nd type of product comes;

The robot will place the first product. For the second box the product will be turned in 270 degrees. When the 3rd type of product comes;

The robot will place the first product. For the second box the product will be turned in 90 degrees.

The program is written with RT toolbox melfa basic 5.0.

The Program Algorithm

Fig. 4: The Algorithm for Test 1

The program code

```

m5=1
*baslat
Wait M_In(4)=1
M_Timer(1)=0
Mov p1
Mov p3,4
Dly 0.5
Mov p4
HClose 1
    
```

```

Mov p5
If M_In(5)=1 Then GoSub *tur1
If M_In(6)=1 Then GoSub *tur2
If M_In(7)=1 Then GoSub *tur3
*algila
If M_In(5)<>1 And M_In(6)<>1 And
M_In(7)<>1 Then
GoTo *algila
Mov p6,10
    
```

```

Dly 0.5
HOpen 1
GoTo *baslat
*tur1
If m5=1 Then
p100=P_Curr ' If it is type 1 it is packed
    with a 180 degrees turn. For the second
    product the turn is 0 degrees
j10=PtoJ(p100)
j10.J6=Rad(180)
p10=JtoP(j10)
Mov p10
m5=2
EndIf
If m5=2 Then
    m5=1
    EndIf
    Return
    *tur2
    .....
    Return
    *tur3
    m5=2
    EndIf
    If m5=2 Then
        m5=1
        EndIf
        Return
        m10=M_Timer(1)/1000
    End

```

Test 2:

The connection diagram of the test:

Fig. 5: Mitsubishi Robot arm with Cognex Camera connection

A covalent system is established with the Cognex Camera, the PC and the Robot Switch.

To work with the Cognex camera in this test, we need:

- Cognex Insight software (software for the camera)
- RT Toolbox (software for the Robotic arm)

- EBTools (Mitsubishi Robot – camera features, a patch program for Ek dll)

And these steps are followed:

- Run the Cognex camera. There is Mitsubishi EBtools on the camera program, so the menu to calibrate the positions of the robot is already in the list.

Fig. 6: Mitsubishi Cognex EB tools settings

In Inspect Part Mitsubishi Robot Tool section the necessary calibration settings are done.

- In camera network settings the camera is introduced and then camera

online option is chosen and continue to work online.

- In RT toolbox online section, the given parameters are changed with the new ones.
- In Robot connection param

No.	Parameter name	Value	Details explanation
1	NETIP	xxx.xxx.xxx.xxx	IP address of robot controller
2	NETTERM(9)	1 (Default value : 0)	The end code is added with communicating by Ethernet.
3	CTERME19	1 (Default value : 0)	The End code of port 10009 is changed to "CR+LF".
4	NETPORT(10)	10009 (Default value : 10009)	Port number allocated to device OPT19
5	CPRCE19	0 (Default value : 0)	The protocol used is "Non-procedure"
6	NETMODE(9)	1 (Default value : 1)	Opens as "Server".
7	NETHSTP(9)	unused	

Fig. 7: Robot parameter settings in Robot- Cognex connection eter,

Fig. 8: Robot connection parametresi

The necessary network settings are defined In Device-Line option for the robot connection.

Fig. 9: Cognex Camera Connection settings

port in robot programming.

The camera network information is done in opt12 section. It is transferred in Com2 section as opt12. **Com2 is the name of the**

- To do the robot calibration a paper mechanism is prepared with a calibration

Fig. 10: Calibrasyon mechanism

- To teach calibration the image of the piece is shot as a camera picture. The image settings (sharpness – brightness) are done. Some points are defined on the image to be the point spots. There can be 4 or 9 calibration points according to the product settlement. After that, all the points are calibrated one by one with the robotic arm board.

Fig. 11: Setting the calibration points

- In our test there are 4 point spots.

	Name	Result
	Point_1	(0.0,0.0)
	Point_2	(0.0,0.0)
	Point_3	(0.0,0.0)
	MCalib_1	Pass

Fig. 12: Point values

the “get position” option.

The positions are got to the point spots with

Fig. 13: Getting the positions to the points

- And then, the chosen position points will be adjusted via Ethernet with string emoticon.

Fig. 14: Settings of the sending Cognex position information via ethernet

Fig. 15: Sending the Cognex point spot parameters

In patterns section, we choose which information will be sent to what points.

The following table is an example.

No.	Display	Explanation
1	Pattern_1.Number_Found	Recognized number
2	Pattern_1.Fixture.X	1st recognition results X
3	Pattern_1.Fixture.Y	1st recognition results Y
4	Pattern_1.Fixture.Angle	1st recognition results θ
5	Pattern_1.Fixture1.X	2nd recognition results X
6	Pattern_1.Fixture1.Y	2nd recognition results Y
7	Pattern_1.Fixture1.Angle	2nd recognition results θ
8	<Same as above>	<Same as above>

- The calibration is done now and teaching the pieces process starts.

To do this process, in locale part section pattern section is chosen and the example closer pieces are defined.

If the close pieces are wanted in settings section, the adjustment is done according to the following table. For example, if the rotation tolerance is 180 degrees the piece will be taught in 180 degrees close images.

Fig. 16 Pattern settings

Here the piece is defined as a screw. The Scara robot will drop the screws into the box next to the platform. In the test, in gripper

option the screw is gripper. Therefore it is dragged.

Fig. 17: Leaving the screw in the test

Experiment was conducted in Mitsubishi Electric Türkiye

Dim PCam(5) ' takes 5 pictures at the same time and the coordinates are kept in the index.

Dim PKAM(5)

*Start

Dly 0.5

If M_NvOpen(1)<>1 Then ' com2: there is an if control (open – close).

NVOpen "COM2:" As #1 'It is the name of the port for the Cognex camera in rttoolbox parameters / connection section.

EndIf

Wait M_NvOpen(1)=1 ' wait till the port is open

*NVld 'Bypass line

NVRun #1, "Arer" 'Arer adlı "JOB" çağır ' It is the position defining file in Cognex. It is run. It is the calibration file.

EBRead #1,

,MNum,PKAM(1),PKAM(2),PKAM(3),PKAM(4),PKAM(5) 'EBRead both asks the camera to take photos and also helps to get the data from the camera. The data is transferred to the PKAM line.

If MNum<1 Then GoTo *Start

For M1=1 To MNum

PCam(M1)=PKAM(M1) ' The string information which come to Pcam line are transferred to the PCam position lines. But when the information comes as X,Y,Z, the A,B,Z values on the transfer point are also transferred.

PCam(M1).Z=Pref.Z

PCam(M1).A=Pref.A

PCam(M1).B=Pref.B

Next M1

'Mov PHOME

For M1=1 To MNum

Mov PCam(M1), 50

Mvs PCam(M1)

Dly 0.1

'Mov PCam(M1), 50

Mov PDrop

Dly 0.5

Mov PDrop, 50

Next M1

Mov PHOME

End

Fig. 18: Algorithm

Conclusion

In this work, robot software tests which are suitable for object definition and flexible manufacturing with Mitsubishi arm and Scara robot were done, using FESTO and COGNEX cameras. In the test results the robot was made to work with image processing. The information of the defined objects were sent to the Mitsubishi robotic arm via Modbus, direct IO outputs and Ethernet protocols. The robotic arm did different movements according to the defined object information. With this study, some experimental software was developed for the robotic arm software which make different movements suitable for the flexible manufacturing. The error rate in the tests was %1.

References

- [1] ARSLAN, F, “Codesys Programlama book” [Codesys programming], Unpublished Master Thesis, Kahramanmaraş Sütçü İmam Üniversitesi , 2012
- [2] KAPTANOĞLU E. “Robotik programlama book” [Robotic Programming], Marmara University, Mechatronics Engineering, 2012
- [3] DEDEOĞLU N. Mitsubishi Electric Robot Programmer, 2014
- [4] ÜNEL M, ”Instruduction Robotics”, Sabancı University 2014
- [5] CRAIG JJ, “Introduction to Robotics book”, 1989,TJ211.C67
- [6] ŞENEL F. “Gömülü sistemler ve görüntü işleme yöntemlerini kullanarak, robot kol ile üretim bandındaki hatalı ürünlerin tespit edilmesi, yüksek lisans tezi” [Using Embedded systems and image processing methods, identifying defective products on the production use robot arms], Unpublished Master Thesis, Süleyman Demirel University,2013
- [7] VURAL M. “Robot Kinematiği” [Robot Kitematic], İTÜ ,2010
- [8] PAPCUN.P, ČOPÍK M. , “Remote control of Mitsubishi industrial robot”,2012
- [9] ERŞAN, E. “Beş eksenli bir robot kolunun çalışma alanı içerisindeki nesneyi görerek algıladıktan sonra bulunduğu konumdan başka bir yere taşınması” [The five-axis robot arm, another place to move from its position after detecting an object in the visual workspace], Unpublished Master Thesis, Marmara University, 2001
- [10] Fang J.,”Four degrees of freedom SCARA robot kinematics modeling and simulation analysis”,2013, International Journal of Computer, (IJ3C), Vol. 2, No.4 (2013)