A Study on Development of Off-Line Path Programming for Footwear Buffing Robot

Tae-Jung Lho*, Dong-Joong Kang*, Woo-Seung Che*, Jin-Young Kim*
and Min-Sung Kim**

* Department of Mechatronics Engineering, Tongmyong University of Information Technology, Busan, Korea
(Tel : +82-51-610-8359; E-mail: tjlho@tit.ac.kr)
** Department of Communication & Information Engineering, Tongmyong University of Information Technology, Busan, Korea

Abstract We suggest how to program off-line robot path along shoes' outsole shape in the footwear buffing process by a 5-axis microscribe system like robot arms. This microscribe system developed consists a 5-axis robot link with a turn table, signal processing circuit, PC and an application software program. It makes a robot path on the shoe's upper through the movement of a microscribe with many joints. To do this, first it reads 5-encoder's pulse values while a robot arm points a shoes' outsole shape from the initial status.

This system developed calculates the encoder pulse values for the robot arm's rotation and transmits the angle pulse values to the PC through a circuit. Then, Denavit-Hartenberg's(D-H) direct kinematics is used to make the global coordinate from robot joint one. The determinant is obtained with kinematics equation and D-H variable representation. To drive the kinematics equation, we have to set up the standard coordinates first. The many links and the more complicated structure cause the difficult kinematics problem to solve in the geometrical way. Thus, we can solve the robot's kinematics problems efficiently and systematically by Denavit-Hartenberg's representation. Finally, with the coordinate values calculated above, it can draw a buffing gauge-line on the upper. Also, it can program off-line robot path on the shoes' upper.

We are subjected to obtaining shoes' outline points, which are 2 outlines coupled with the points and the normal vector based on the points. These data is supposed to be transformed into *.dxf* file to be used for data of automatic buffing robot. This system developed is simulated by using spline curves coupled with each point from *dxf* file in *Autocad.* As a result of applying this system to the buffing robot in the flexible footwear manufacturing system, it can be used effectively to program the path of a real buffing robot.

Keywords: off-line robot path programming, footwear buffing robot, FMS, 5-axis microscribe system, Denavit-Hartenberg's direct kinematics

1. Introduction

In the present footwear industry, people operate machines or work by themselves in most processes. Thus, the industry is going down because of reduction of productivity and avoidance of people. Among the footwear processes, a process for bonding outsole and the upper leather or making a complete product has a low working environment and has the frequent workmen's accident. But the most significant reason is the features of shoe outsole. Since the outsole has many kinds in size, shape, color, and raw materials, it is not only difficult to actively develop a system coping with diversity, but also inefficient to economics**[1][2]**.

In the recent researches for developing of flexible footwear manufacturing system, they have been conducted off-line path programming system to be used for buffing process, and bonding process in footwear assembling facilities by the research and development part. However, these researches are the first step, so it can't keep pace with the trend of automatic facilities for the flexible manufacturing system.

The developed countries are making an effort to focus on production and innovation of items at the same time to distinguish from other developing countries. Also, those developed countries intensively invest to FMS(Flexible Manufacturing System). In addition to that, to develop an innovative product, the countries try to use new scientific human technology and environment - to a shoe drive forward the innovation of a product and manufacture at the same time. Therefore, it is necessary for our country to introduce FMS and an automatic facility only for shoes corresponding to the domestic situation.

In this study, as the 5-axis link robot with the turn table moves along a processing path on the shoes' upper, the off-line programming system generated the processing path of the shoe's buffing robot and transferred them to its controller **[3]**.

2. System Configuration

This system developed calculates the encoder pulse values for the robot arm's rotation and transmits the angle pulse values to the PC trough a circuit. Then, Denavit-Hartenberg's (D-H) direct kinematics is used to make the global coordinate from robot joint one. The determinant is obtained with kinematics equation and D-H variable representation. To lead the kinematics equation, we have to set up the standard coordinates first. The many links and the more complicated structure cause the difficult kinematics problem to solve in the geometrical way. Thus, we can solve the robot's kinematics problems efficiently and systematically by Denavit-Hartenberg's representation.

We can display the shoe's outsole bonding part shape by using the 5-axis, and create the auto-bonding robot's moving coordinate by the normal vector. There is the whole system configuration in figure 1. Since 5-axis multi-joints microscribe easily and fast digitizer a model, if a user makes probe track outline of an object, complex 3-dimension data are created in the computer. Because of fast digitizing an real object, the microscribe is a dialog measuring device in 3-dimension computer environment. Also, it has a merit that it digitizer all objects without concerning shape, size, and materials **[4]**.

The shoe's outsole is marked by a probe of the microscribe, and the encoders' pulse values are read and counted in the *PCL-833 encoder pulse counter* board from the origin point to the probe's location.

Fig. 1 System configuration

Then, we can obtain the coordinate by the kinematics and display the shoe's outsole on the screen through C-program. At *PCL-833 encoder pulse counter*, we can count in 24bit with 3-axis encoder counter card **[5]**.

Pulses from the robot arm's encoders go to the *PCL-833 encoder pulse counter* board. First, count pulse numbers at the *PCL-833 encoder pulse counter* board, and then, transform the counted pulse into the radian value in the program. That is, if angle for a pulse is known value, we can obtain angle to the total pulse numbers. With the radian values from the pulse angles and the known link length, the last position (shoe's outer point location from the origin point) can be extracted from the direct-kinematics. Then, the T/T calculates rotation and translation coordinates from the 5-axis direct-kinematics freely rotating and marking points, and T/T pulse values (radian value). Because the shoe's outer coordinates are displayed on the screen, it is able to obtain the normal vector from shoe's 3 points, and then, save this normal vector as text file. The normal vector saved as a text file is used for obtaining robot's processing paths for the buffing process.

Figure 2 is represents the process for obtaining a coordinates and figure 3 shows a overview of the microscribe developed here.

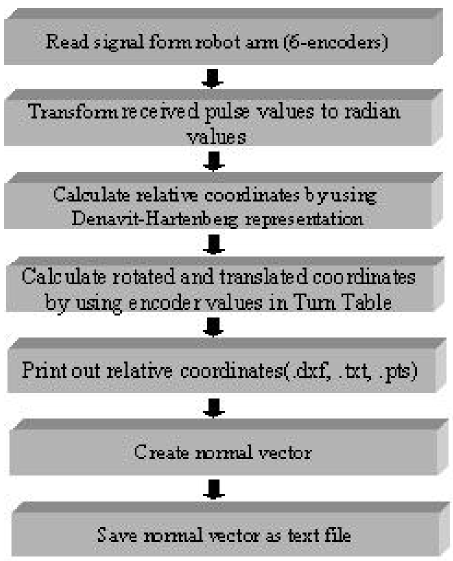

Fig. 2 The process steps for obtaining a coordinates

Fig. 3 Overview of a microscribe with 5-axis arm

3. Denavit- Hartenberg's(D-H) Direct Kinematics

In the global coordinates (x, y, z) after extracting a value form the micrscribe. The determinant is obtained with kinematics equation. We have to set up the standard coordinates first. The many links and the more complicated structure cause the difficult kinematics problem to solve in the geometrical way. Thus, we can solve the robot's kinematics problems efficiently and systematically by Denavit-Hartenberg's representation **[6][7][8]**.

From the coordinate of 5-axis robot shown in figure 4, the Denavit-Hartenberg's variables extracted from the set-up coordinate are arranged and shown in the below Table1.

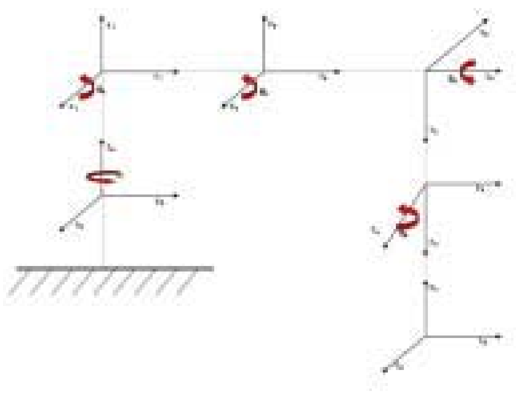

Fig. 4 Coordinate of 5-axis robot manipulator

Table 1 Denavit-Hartenberg's variables of each axis.

Link	θ_i	d_i	α_i	a_i
A_1	θ_1*	d_1	90	0
A_2	θ_2*	0	0	a_2
A_3	θ_3*	0	−90	0
A_4	θ_4*	d_4	90	0
A_5	θ_5*	0	0	a_5

To solve all kinematics, we can work on by setting up a coordinate at each link as one pleases. But to effectively solve kinematics, it is useful to set up a well organized. The most frequently used way is Denavit-Hartenberg's(D-H) representation. We can obtain the global coordinates by transforming pulse numbers to values and calculating. The below determinant is 5-axis robot's calculation in the direct kinematics. Each link matrix is obtain with inputting the variables in Table 1 to the below determinant.

By D-H representation, Transform homogeneous transform A_i into multiply of 4 normal transform matrices **[9]**.

A_i = Rotz, Θ_i Transz,d_i Transx,a_i Rotx, α_i

$$= \begin{bmatrix} C\theta i & -S\theta iC\alpha i & S\theta iS\alpha i & aiC\theta i \\ S\theta i & C\theta iC\alpha i & -C\theta iS\alpha i & aiS\theta i \\ 0 & S\alpha i & C\alpha i & di \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$${}^{5}_{0}A = A_1 A_2 A_3 A_4 A_5 \quad (2)$$

The obtained 4x4 homogeneous transformation matrices are shown the below, which indicates relative location and direction of set-up coordinates $\{i\}$ and $\{i+1\}$. Finally the homogeneous transformation matrix is as follows:

$${}^{0}_{5}A = \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

where,

$$r_{11} = c_5(c_1c_{23}c_4 - s_1s_4) - c_1s_{23}s_5$$
$$r_{12} = -s_5(c_1c_{23}c_4 - s_1s_4) - c_1s_{23}s_5$$
$$r_{13} = c_1c_{23}s_4 + s_1c_4$$
$$r_{21} = c_5(s_1c_{23}c_4 + c_1s_4) - s_1s_{23}s_5$$
$$r_{22} = -s_5(s_1c_{23}c_4 + c_1s_4) - s_1s_{23}s_5$$
$$r_{23} = s_1c_{23}s_4 - c_1c_4$$
$$r_{31} = s_1c_{23}s_4 - c_1c_4$$
$$r_{32} = -c_4s_{23}s_5 + c_{23}c_5$$
$$r_{33} = s_{23}s_4$$

,and

$$p_x = a_5a_{00} - c_1s_{23}d_4 + c_1c_2a_2$$
$$p_y = a_5a_{10} - s_1s_{23}d_4 + s_1c_2a_2$$
$$p_z = a_5a_{20} + c_{23}d_4 + a_2s_2 + d_1$$
$$a_{00} = c_5(c_1c_{23}c_4 - s_1s_4) - c_1s_{23}s_5$$
$$a_{10} = c_5(s_1c_{23}c_4 + c_1s_4) - s_1s_{23}s_5$$
$$a_{20} = c_4s_{23}c_5 + c_{23}s_5$$

We describe how to display a shoe's shape followed by the direction of the turn table (T/T). The reason for the rotation of the T/T is that it is difficult for a robot arm to measure when a long-length shoe outsole or a big item is measured. By using the T/T, we can not only have more space, but also obtain more accurate point through rotating the T/T as 90° or 180°. As shown in figure 5, first, move parallel to overlap the T/T's axes and a robot arm's axes. Then, rotate as much as assigned, and the axis is moved parallel back to the origin position

$$(\,x'\ y'\ z'\ 1\,) = (\,x\ y\ z\ 1\,)\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ a-0 & b-0 & c-0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} \cos\Xi & \sin\Xi & 0 & 0 \\ -\sin\Xi & \cos\Xi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0-a & 0-b & 0-c & 1 \end{pmatrix} \quad (4)$$

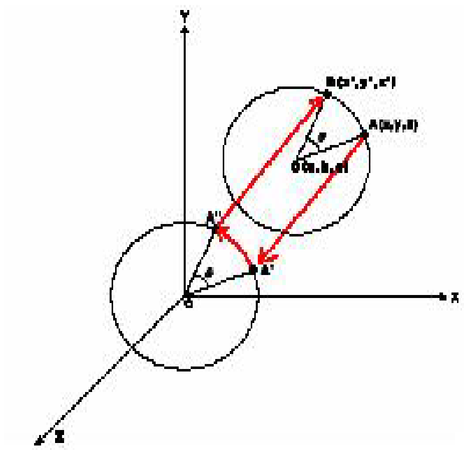

Fig. 5 Transformation of turn table rotation

4. The extraction of shoe's outsole coordinates

A drawing of the shoe's outsole is necessary to draw the outline of the shoe's outsole with *AutoCAD* or *CATIA*,. But if we use the 5-axis robot, we can draw the outline of the shoe's outsole without the drawing of the shoe's outsole. We can cover how the automatic bonding robot finds the shoe-bonding paths based on the shape like the figure 3.

As a shoe fixed on the T/T are rotated and is measured by a 5-axis robot moving along the outline of the shoe, 6 encoder values are saved into a computer's memory through *PCL-833 encoder pulse counter* board. The saved encoder values are transformed into Degree from Radian, are calculated for 3-dimensional coordinates (*x, y, z*) by the D-H representation, and obtained coordinates are displayed on the screen as a shoe's winding shape itself. Figure 5 shows the process to display a real shoe outsole shape fixed on the T/T. With obtained coordinates through figure 6, there is a coordinate extraction program for a shoe shape in figure 7 **[10]**.

Fig.6 Overview for working by microscribe

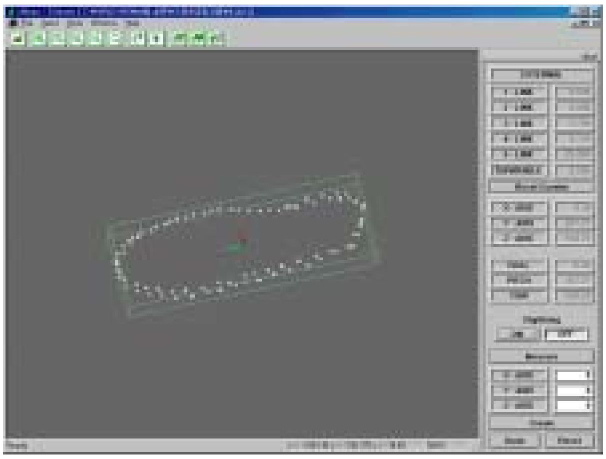

Fig.7 Creation for the outer coordinates

The probe pointed around 80 points along a buffing outline. In order to obtain the normal vector, we measured the up and down lines. To compute the normal vector, 3 coordinates are needed. The probe has to point out along the up and down lines to have the normal vector at each point in the two lines.

These points are saved as a *txt* file to compute the normal vector through a normal vector generator. The normal vector generator program is that it, first, reads previously marked points, creates the normal vector, throws away the first point, re-reads a new point, and creates the normal vector. It generates each normal vector at each point until it reaches the saved last point. Table 2 shows each point's coordinate of a shoe outline as text file. There is a generated file in the form of the vector from the Table 3's text file. In this file, the location of a point is shown as '*mm*' unit from the origin of the robot arm. That is, the robot arm's initial state is (*x, y, z*)=(0, 0, 0) and from the initial state, it shows point coordinates for a shoe outsole. It helps to figure out a present coordinate correctly marked..

Table 2 Text file of points coordinate

-431.440	-68.860	183.670
-431.530	-68.960	184.140
-428.080	-43.850	178.660
-425.010	-23.790	179.220
-425.180	-12.030	192.130
-419.530	-7.080	179.720
-419.890	5.810	190.170
-416.340	12.110	179.220
-415.400	46.960	172.660
-414.250	45.940	172.680
-416.800	58.590	168.440
-416.190	58.050	168.030
-419.990	69.190	176.160
.		
.		
-424.420	-114.580	175.720
-424.420	-114.580	175.720
-431.340	-105.350	184.600
-424.490	-99.660	175.300
-431.760	-89.030	182.000
-424.820	-83.170	173.840
-423.520	-83.440	174.290
-432.230	-71.010	181.240
-427.460	-65.070	174.830
-427.660	-64.260	174.400

Table 3 Text file of normal vector

-0.981	0.098	-0.167
0.987	-0.149	-0.063
0.976	-0.154	0.153
-0.907	0.306	-0.291
0.916	-0.236	0.323
-0.962	0.154	-0.223
0.944	0.036	0.327
-0.142	-0.179	-0.973
0.301	0.357	0.884
-0.715	-0.347	-0.606
0.030	-0.582	0.812
-0.964	-0.227	-0.140
0.987	0.126	0.103
	.	
	.	
-0.339	-0.590	0.733
0.729	0.619	-0.294
0.674	0.338	0.657
-0.799	-0.021	-0.601
0.763	0.121	0.636
-0.733	-0.075	-0.677
0.026	-0.822	-0.568
-0.367	-0.636	0.678
-0.730	-0.137	-0.670
0.400	0.505	0.765

Table 4 DXF file of points

0
SECTION
2
ENTITIES
0
POINT
5
0
8
0
10
-431.440
20
-68.860
30
183.670
.
.
-427.660
20
-64.260
30
174.400
0
ENDSEC
0
EOF

We are subjected to obtaining shoes outline points, 2 outlines coupled with the points, and the normal vector based on the points. These data is supposed to be transformed into *.dxf* file to be used for data of automatic buffing robot. This system developed is simulated by using spline curves coupled with each point from *dxf* file in *Autocad* **[11]** in Table 4.

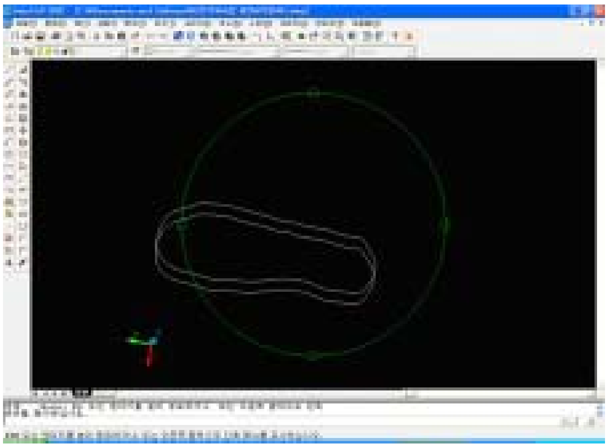

Fig.8 Shoe outsole in 3-dimension

Figure 9 shows what the normal vectors of a shoe outsole are saved into a robot and applied to real show manufacturing line's robot in the off-line.

Fig.9 Application to the buffing robot

5. Conclusion

In this paper, automatic bonding path how to obtain the path for the automatic bonding robot to move is subjected. Shoe outsole bonding area is created after inputting a shape from 5-axis robot and is displayed on the screen at the same time. Then, it is saved as a file. These whole steps are the processes how to make auto bonding robot's path.

When a man does the buffing job at non-factory automation system, it has merits for the speed of processing time, but it also has demerits for accuracy. Buffing process line is depended on responsible people's sense and expertness. Bonding process line has the same story. Therefore, finished shoes have bonding area flowed out from outsole and are separated from outsole in no time.

However, in the case of installing buffing robot in the shoe product line, it takes around 11-12mm/sec when shoe's length is 260mm. And it takes about 30 seconds to move a pallet point to work space of buffing robot from conveyer belt. In addition to that, it takes about 1 minute for the process. Thus, the whole process time is about 1 and half minutes for each shoe. That is to say, the time is for getting through the process of buffing robot. The next bonding process is also needed the similar time of buffing process. The bonding robot accurately bonds along the exact line of buffing robot.

This study is a stage of shoe factory automation, and is expected to reduce labor cost and to improve the quality of products. In addition, by helping automation of shoe industry, this report can contribute to the improvement of productivity. As a result of applying this system to the buffing robot in the flexible footwear manufacturing system, it can be used effectively to program the path of a real buffing robot.

Acknowledgments

This study was supported by the financial aid of Nanux Co.Ltd and Ministry of Commerce, Industry and Energy in Korea

REFERENCES

[1] H. J. Ahn and D. H. Kang "Growth flow of shoe outsole's materials for injection", *Rubber Technology* Vo1.1, No.2, 2002.

[2] D. H. Park, D. H. Moon and C. U. Lee, "Study on Bond Used in Shoes Manufacturing Industry", *J. of Korean Industry Hygiene Association*, Vol.1, No.2, pp.200-213, 1991.

[3] J. Y. Kim, D. J, Kang, T. J. Lho, H. K. Ahn and H. S. Cho, "Robot Off-Line Programming Based on CAD Data in Adhesive Application System for Shoes Outsoles and Upper", *Proc.of ICCAS2002*, pp. 708-712, 2002.

[4] Ashinuma Retz and Narui Hiroshibu, "Laser digitizer and 3-dimension CAD/CAM Application", *Machine Design*, pp 41-44, 2000.

[5] *3-axis quadrature encoder and counter card user's manual*, Technologies Advantech Co. Ltd. 1994.

[6] R. J. Schilling, *Fundamentals of Robotics-Analysis and Control"*, Prentice Hall Ed., 1990.

[7] J. G. Kermas, *Robot Technology Fundamentals*, SciTech Media, 2002.

[8] T. J. Lho, D. J. Kang, W. S. Che, S. H. Song and S. J. Cho, "A Study on Off-Line Programming of Robot Path for Footwear Bonding Automation", *Proc. of ICCAS2002*, pp.2066-2069, 2002.

[9] Y. C. Park and K. Y. Kim "Kinematics analysis of 6 axis robot for industry", Seoul National University of Technology the correction of learned paper 46th, pp223-235, 1997.

[10] S. Y. Lee, *Visual C++ Programming Bible(Ver.6.)*, YoungJin Pub.Co., 2001.

[11] J. S. Um, *AutoCAD 2000 Bible*, YoungJin Pub. Co., 2000.