International Journal of Fuzzy Logic and Intelligent Systems, vol. 12, no. 3, September 2012, pp. 256-262
http://dx.doi.org/10.5391/IJFIS.2012.12.3.256 pISSN 1598-2645 eISSN 2093-744X

Post-processing of Direct Teaching Trajectory in Industrial Robots

Taeyong Choi†, Chanhun Park, Hyunmin Do, Kwangcho Chung, Dongil Park and Jinho Kyung

Department of Robotics and Mechatronics, Korea Institute of Machinery& Materials, 156, Gajeongbuk-Ro, Daejeon, 305-343, Korea

Abstract

Direct teaching of the industrial robot is a novel technique to easily teach manipulators. However, teaching data by human hand cannot help having large noise errors ranging from low to high frequency. To use teaching data, post-processing to correct the teaching trajectory is required. Here, a novel shape-based trajectory correction method to rebuild teaching data with the feature information of curvature and velocity is proposed. The proposed method is tested on square and circular objects.

Key Words: Direct teaching, Trajectory correction, Feature point extraction.

1. Introduction

Industrial articulated robot arms are usually used for factory automation. In general, industrial robot arms carry out operations based on the teaching-playback method. This method is separated into two parts, i.e., the teaching part and the playback part. The conventional method to teach the manipulator was with a complex computer and teaching-pendant [1, 2]. Recently, some researchers proposed novel robot teaching methods [3–7] including the direct teaching method. Direct teaching (DT) is one of the intuitive teaching methods. An operator pushes or pulls the end-effector of a manipulator and it is complies with the operator's teaching force/moment; as a result, it moves as the operator intends. The manipulator records the teaching trajectory and plays it back at the operator's request. This method is very easy and effective compared to conventional methods like teaching by teaching-pendant. However, there are also many drawbacks to DT, i.e., teaching data extraction, teaching trajectory correction, trajectory optimization, etc. Among these, teaching data extraction and correction are considered here.

Teaching data extraction refers to shape data extraction without the approaching trajectory, that is, the trajectory from the origin position to the beginning position of the target shape, or the departing trajectory, that is, the trajectory from the end position of the shape trajectory the to origin position. Teaching data correction means modifying the trajectory taught by the human to be a more correct trajectory, suitable for use by the robot. The DT trajectory is distorted through the data gathering process by F/T sensor, ADC noise, and human hand error [8]. Human hand error is a significant problem. Drawing a straight line on the note with a pen is almost impossible for a human. Similarly, drawing a straight line on the material with a robot handle is actually impossible. This noise problem is not significant for the approaching and departing trajectories. However, it is critical for the shape trajectory.

Correction of the shape trajectory requires removal of various types of noise. Human hand error cannot actually be modeled numerically. Elimination of human hand error is dealt with in the free-hand sketch recognition on the handheld PC or tablet [9]. Sezgin *et al.* used curvature and velocity data to correct 2-dimensional freehand sketches. However, their work is not fit for precise operation of industrial robots.

Here, a novel shape-based trajectory correction (STC) method is proposed. DT trajectories are extracted from all trajectories of the robot. The extracted shape trajectories which contain both position and orientation are corrected based on shape characteristics, to be more suitable for robot operation..

2. Feature Point Recognition

We built a manipulator with the direct teaching function as shown in Fig.1. A manipulator is guided along the outer edge of the teaching area for deburring work. Deburring is a good example for DT application [10] because it is a tedious and repetitive job. The robot that we developed contains a 6-axis F/T sensor at the end-effector to measure contact force as shown in Fig.2. The basic purpose of this

Manuscript received Nov. 7, 2011; revised Jun. 7, 2012; accepted Aug. 12, 2012.

†Corresponding Author : taeyongc@kimm.re.kr

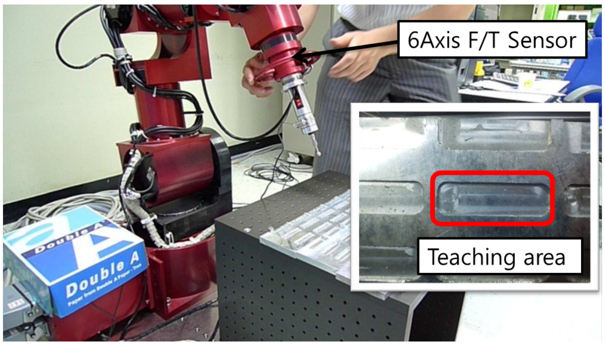


Figure 1: Direct Teaching Robot with 6-axis F/T sensor being embedded in teaching module.

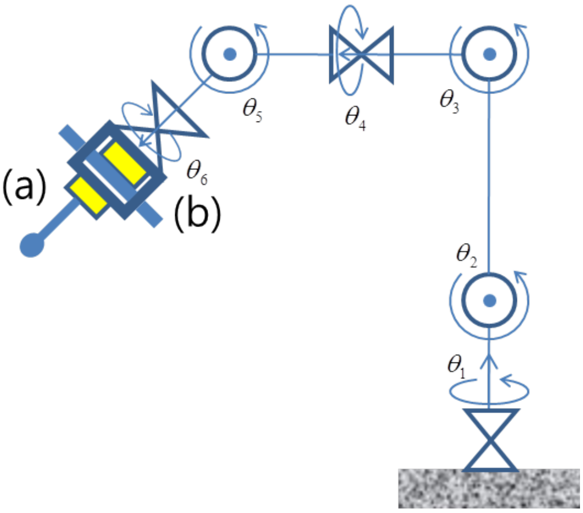


Figure 2: The manipulator have two 6-axis F/T sensors such as (a) and (b). (a) F/T sensor used to detect teaching start and end times.

sensor is to measure the contact forces precisely to enhance the impedance control performance during direct teaching. The F/T sensor input from Fig.2-(a) are actively used to extract shape trajectory only.

2.1 Teaching data acquisition

A user moves the handle of the manipulator from the origin to the teaching start point on the plate and then returns it to the origin. Fig.3-(a) shows that all trajectories of the end-effector consist of the approaching trajectory, shape trajectory, and departing trajectory. Measured F/T sensor input are plotted versus time as the black line in Fig.3. Among the six measured F/T sensor inputs, only the sum of the x-y-z directional forces is used. Using only the z-directional force is not enough because teaching in 3D space has to be considered. So, we add x, y, and z directional forces as in (1).

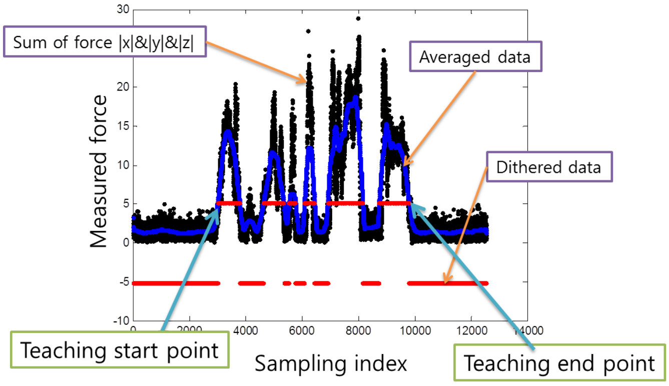


Figure 3: Filtered teaching data are estimated. Averaging filter(blue), dithering(red) are applied.

$$MF(i) = force_x(i) + force_y(i) + force_z(i) \qquad (1)$$

MF is the measured force, and i is the sampling index. The sum force MF is filtered with averaging as in (2). For the data $i < window/2$ and $i > I - window/2$, i or $I - i$ are selected as a half of window length. Here, I is the total sampling number.

$$AMF(i) = \sum_{k=i-window/2}^{i+window/2} MF(k)/window \qquad (2)$$

AMF means averaged MF.

$$\begin{aligned} \forall i,\ if\ AMF(i)\ &< Mean(AMF) \times \eta, \\ Then, AMF(i) &= -\varphi \\ Otherwise, AMF(i) &= \varphi \end{aligned} \qquad (3)$$

The averaged filter data is dithered as in (3). η in (3) is a division constant, which was set to be 1 in experiments. φ in (3) is a user parameter. Finally, the processed force data is plotted as the red line in Fig.3. In Fig.3, the dithered plot (red lines) is not high for all teaching trajectory. There are many valleys between the start and end points of teaching. This is because of human error. It is likely that the human operator will hold the teaching handle loosely when turning his attention to others. To extract only the start and end points of teaching, all positive zero-crossing and negative zero-crossing points for the dithered data are detected. The first positive zero-crossing point is selected as the teaching start point, and the last negative zero-crossing point is selected as the teaching end point.

Finally, the extracted teaching trajectory is shown in the blue line in Fig.4. The teaching trajectory including approaching and departing is actually displayed in 3-Dimensional space. Fig.4-(b) shows the top view and (c) shows the side view. The left two black lines are the approaching and departing trajectories.

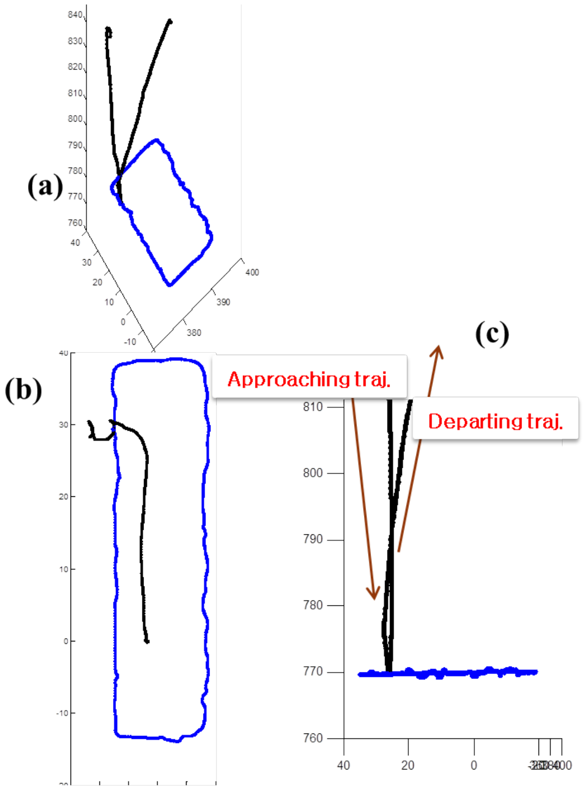


Figure 4: A teaching trajectory, including approaching and departing trajectory, is shown. The blue lines of (a), (b) and (c) are the extracted teaching data. Unit(X-axis, Y-axis):mm.

2.2 Sampling data number reduction

Teaching a manipulator is usually a slow process. Inevitably dummy or duplicate data are sampled at a constant sampling rate. This is critical to the necessary computing power for a feature recognition algorithm followed. More than 50% of the sampling data was dummy data in our experiments. To remove that dummy data, Douglas-Peucker Algorithm (DPA) is used with low threshold [11–13]. Actually, DPA is used to smooth the line with the proper user parameter, which also smooths the corner points. Here, corner points are very important to correct the direct teaching data. Corner points must be detected to recognize the feature points correctly, so the DPA with very small threshold is used to avoid having an effect on the corner points of the original trajectory. Only, dummy data is removed with the DPA.

2.3 Morphological Feature points

The DT trajectory should be corrected for the robot. The original profile contains various noises. First, high frequency noise makes the robot need a very high speed end-effector movement capability. Second, low frequency noise due to human teaching error causes reduced performance. The proposed STC method use the curvature and inverse of the velocity to recognize the feature points of the DT trajectory. Shape trajectory is corrected with the extracted features. Curvature is the inverse of the radius of the circle which contains neighborhood points. Curvature is high at the corner points, and it is low on the line segments. Using this property, segments of the trajectory are classified as straight lines, corners, and circle segments. To remove the low frequency noise from human teaching error, the Gaussian kernel of (4)is convolved with the curvature. The Gaussian has two parameters: median and deviation. We change the deviation. High deviation removes more information, even true teaching information, and low deviation removes less information, leaving the human-hand error. So the selection of the proper deviation is important. Fig.6 shows the number of peaks at a specific deviation of the curvature. We select the incline-changing point as proper deviation. Most of high frequency noise are removed at this time. The number of peak points are decreased with high inclination until the high frequency noise is abolished. After that, the inclination is decreased. That inclination variation point is selected. Fig.5 shows the variation of the feature points as the deviation increases. Some feature points are rejected as the deviation increase. Surviving points are considered true feature points. For each result of the Gaussian convolution of curvature and velocity, proper deviation values are chosen like PD in Fig.6 and Fig.8. Fig.11 shows the original curvature and inverse velocity plot with the selected feature points at the proper deviation. Selected features are plotted in Fig.10

$$f(x) = e^{(-\frac{(x-c)^2}{2\sigma^2})} \tag{4}$$

where c is the center of the Gaussian function σ is the deviation. Features from the curvature and velocity need to be ordered to select a proper feature set to describe the morphologically significant points in the shape profile. Feature points from the curvature and velocity are merged based on the orthogonal distance from the trajectory point to the relevant line segment of the polyline as in (5). For all feature of the curvature and velocity, the feature set F and its errors with respect to the original shape trajectory are decided through Algorithm 1 and plotted in Fig.9. After a certain time, the error stops decreasing suddenly. The rule to select the choosing point which cut the feature points to configure the shape trajectory is the variation of slope. For example, the orthogonal distance between a simple square trajectory and three feature points can be extremely larger than that between a simple square trajectory and four feature points. The slope of errors varied suddenly when the fourth feature is added. The surviving points are selected as a proper feature set.

$$\varepsilon(F_i) = \frac{1}{|S|}\sum_{s\in S} OD(s, F_i) \tag{5}$$

where F_i is a feature set from Algorithm 1. Index i is also from Algorithm 1. OD means the orthogonal distance from a feature set to the original trajectory. S is the DT trajectory length.

Algorithm 1 Hybrid feature selection

```
procedure FEATURE MERGE($C, V$)
                    ▷ C:Curvature feature, V:Velocity feature
    $F_0 \leftarrow C \bigcap V$
    $CL \leftarrow C - F_0; VL \leftarrow V - F_0$
    for $All\, CL\, and\, VL$ do
        $i \leftarrow i + 1$
        $C_{MAX} \leftarrow Max(CL)$
        $V_{MAX} \leftarrow Max(VL)$
        $F_{i_c} \leftarrow F_{i-1} \bigcup C_{MAX}$
        $F_{i_v} \leftarrow F_{i-1} \bigcup V_{MAX}$
        if $\varepsilon(F_{i_c}) < \varepsilon(F_{i_v})$ then        ▷ $\varepsilon$ of (5)
            $F_i \leftarrow F_{i_c}$
            $CL \leftarrow CL - C_{MAX}$
            $VL \leftarrow VL$
        else
            $F_i \leftarrow F_{i_v}$
            $CL \leftarrow CL$
            $VL \leftarrow VL - V_{MAX}$
        end if
    end for
end procedure
```

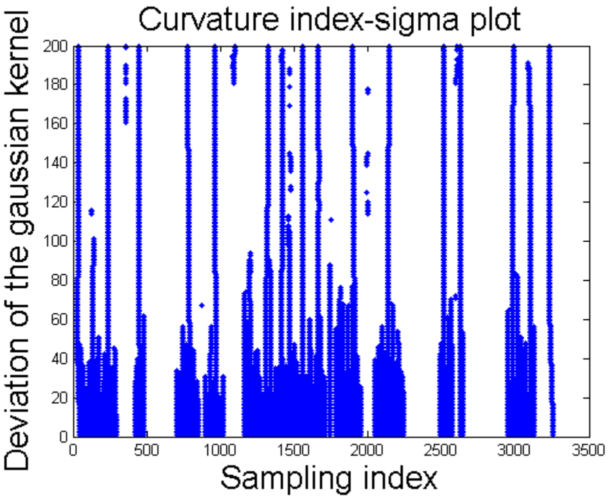


Figure 5: Horizontal axis is sampling index during direct teaching. Vertical axis is the gaussian kernel deviation applied to the curvature data.

2.4 Teaching trajectory correction

Using the selected feature set, the shape trajectory is reconfigured. Here, each segment is divided into features which are distinguished as either curves or lines. The measure for distinguishing curves and lines for each segment consisting of consecutive feature points is the ratio of Euclidian length to accumulated length as in (6).

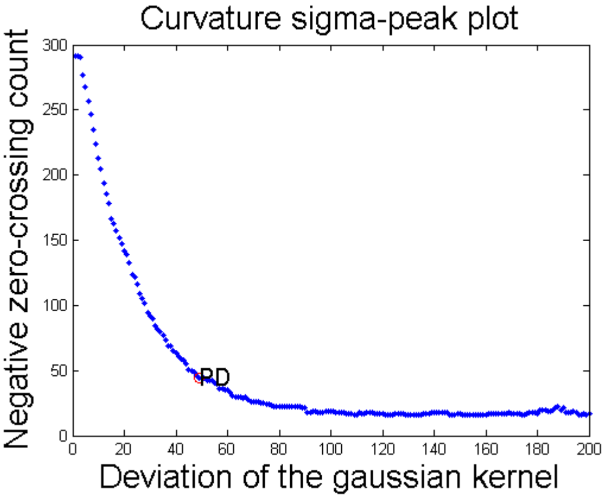


Figure 6: Horizontal axis is the gaussian kernel deviation applied to the curvature data. Vertical axis the the number of the peak points. PD means proper deviation.

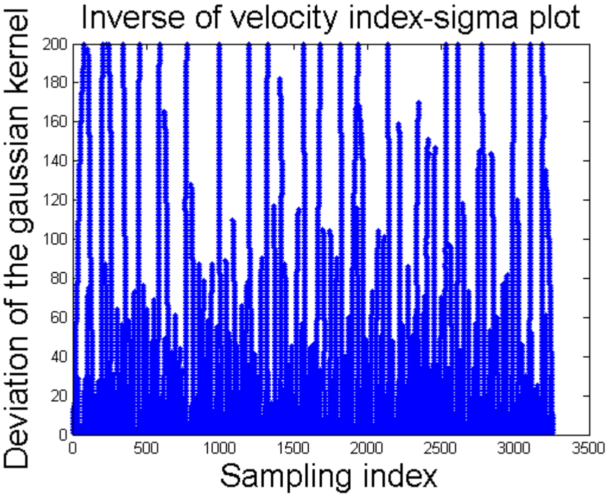


Figure 7: Horizontal axis is the gaussian kernel deviation applied to the curvature data. Vertical axis the the number of the negative zero-crossing.

$$\delta = \frac{Euclidian\, length(s_i, s_{i+1})}{accumulated\, length(s_i, s_{i+1})} \qquad (6)$$

where if $\delta > \gamma$, the user-defined parameter, that segment is considered as a straight line, otherwise it is a curve. γ is always less than 1. The curve is reshaped as a simple $B\acute{e}zier$ curve.

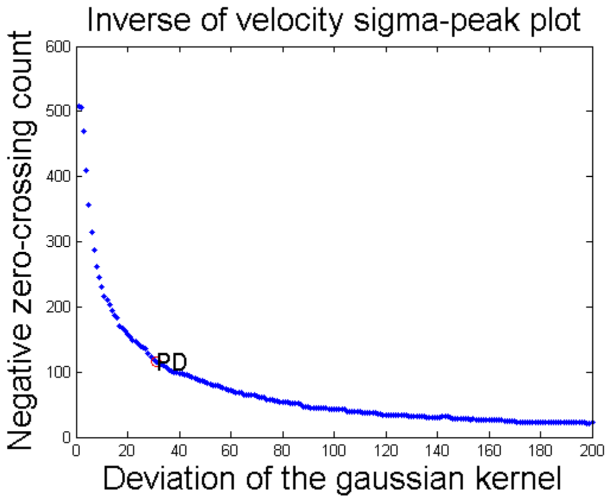


Figure 8: Horizontal axis is the gaussian kernel deviation applied to the curvature data. Vertical axis the the number of the negative zero-crossing.

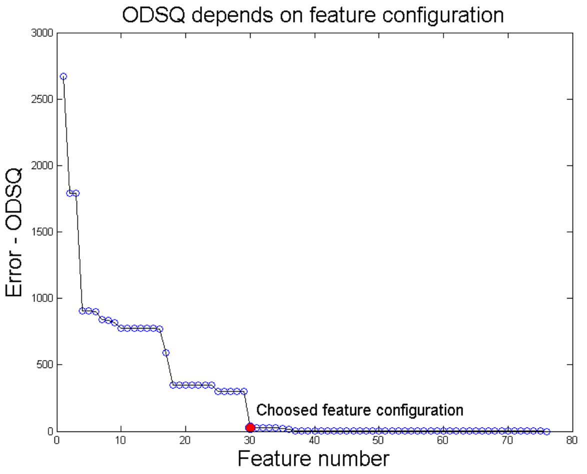


Figure 9: ε of the feature set from the algorithm 1 are plotted. In certain time, decrease of error stop suddenly. That point is selected proper feature set.

3. Experiment

The developed STC method is tested for the square and circular shaped objects. Experimental results are compared with those of the DPA to prove the superiority of the STC. The target objects are shown in Fig.12 The mean squared error is used to compare the real trajectory of the object and the resulting trajectory of the algorithm used. Table.1 shows the result of experiments. Fig.13 shows the trajectory of the STC and DPA. The STC shows good results compared to the DPA. The STC is relatively weak for a circular object because it is based on the feature of corner or edge points. However, the STC is better than the simple

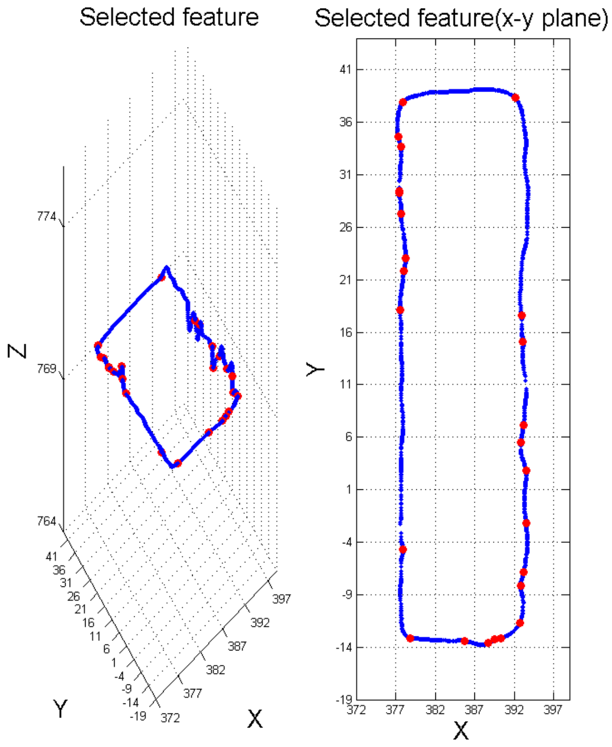
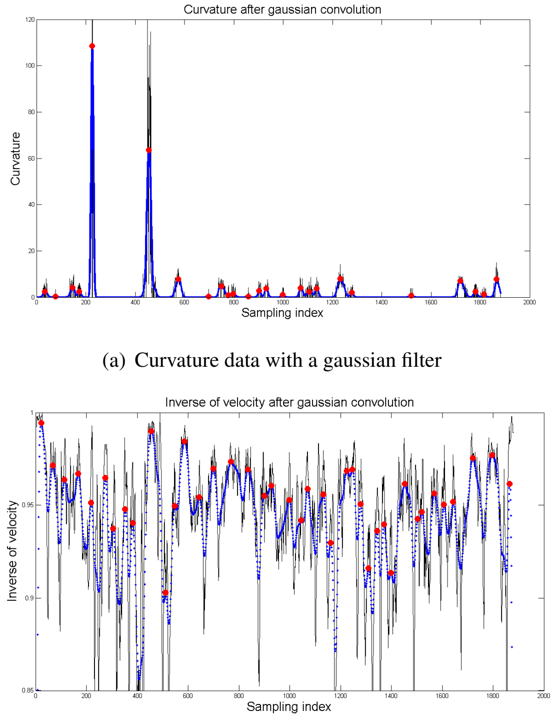


Figure 10: Feature points of curvature plot and velocity plot are listed in the 2D shape profile. Unit(X-axis, Y-axis):mm.



(a) Curvature data with a gaussian filter

(b) Inverse of velocity data with a gaussian filter

Figure 11: (a) and (b) show the selected feature points at a proper deviation value on the curvature plot and the inverse of velocity plot.

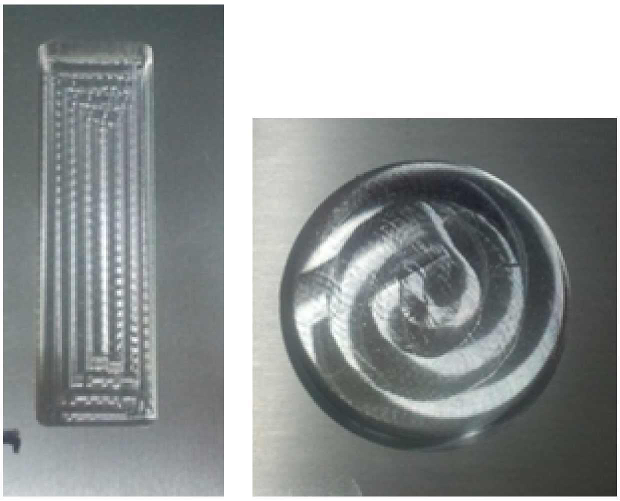
Table 1: MSE for each experiments. Error unit : mm

	Square	Circle
DPA	0.25	0.19
STC	0.12	0.14

line smoothing method like the DPA used in the experiments. Also, the STC enhances the smoothness of the line without smoothing the corner points.

4. Conclusion

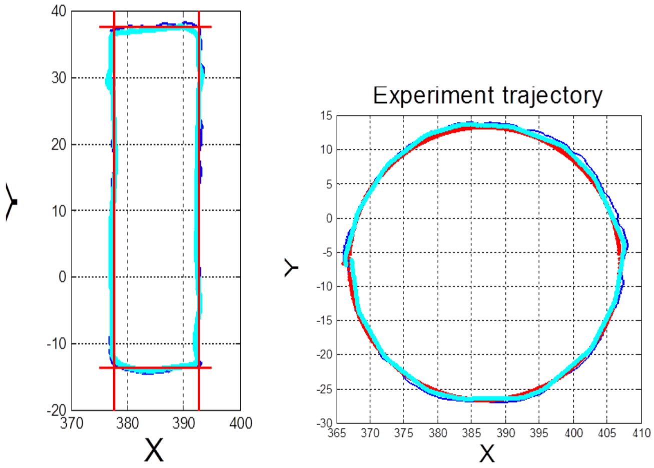
The shape trajectory of the DT trajectory was corrected based on curvature and velocity features. It shows a smooth and simplified shape trajectory. Using it, the robot can move more naturally with no overload to the controller. However, it shows weakness in circular shape. To advance the proposed method, extending the algorithm to consider the circular trajectory will be required. Also, comparison studies with other similar methods on various objects are planned.



(a) Square object

(b) Circle object

Figure 12: Target object. (a) square object(width=15mm, height=50mm). (b) circular object(radius=20mm)



(a) Square shape trajectory

(b) Circle shape trajectory

Figure 13: The red line is the object trajectory. The blue line is the trajectory from the DPA. The cyan line is the trajectory from the STC. Unit(X-axis, Y-axis):mm.

References

[1] Y. F. Yong, J. A. Gleave, J. L. Green, and M. C. Bonney, “Off-line programming of robots,” *Handbook of Industrial Robotics*, S. Y. Nof, Ed. Wiley, 1985, pp. 366-380.

[2] Y. Shamash, Y. Yang, and Z. Roth, “Teaching a robot,” *International Encyclopedia of robotics: Applications and Automation*, R. C. Dorf, Ed. Wiley, 1988, pp. 1689-1701.

[3] C. H. Park, J. H. Kyung, D. I. Park, K. T. Park, D. H. Kim, and D. G. Gweon, “Direct Teaching Algorithm for a Manipulator in a Constraint Condition using the Teaching Force Shaping Method,” *Advanced Robotics*, no. 24, pp. 1365-1384, 2010.

[4] S. Jeong, Y. Nakabo, T. Ogure, and Y. Yamada, “Development of a direct teaching system for a cooperative cell-production robot considering safety and operability,” *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, pp. 5339-5344, Oct. 2009.

[5] Y. Maeda, T. Ushioda, and S. Makita, “Easy robot programming for industrial manipulators by manual volume sweeping,” *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 2234-2239, May. 2008.

[6] A. Stopp, S. Horstmann, S. Kristensen, and F. Lohnert, “Toward interactive learning for manufacturing assistants,” *IEEE Trans. on Industrial Electronics*, vol. 50, issue.4, pp. 705-707, 2003.

[7] G. Grunwald, G. Schreiber, A. Albu-Schaffer, and G. Hirzinger, “Programming by touch: the different way of human-robot interaction,” *IEEE Trans. on Industrial Electronics*, vol. 50, issue 4, pp. 659-666, 2003.

[8] D. I. Park, C. H. Park, and J. H. Kyung, “Signal processing of direct teaching data for human-robot cooperation,” *Proc. Int. Symposium on Industrial Electronics*, pp. 1956-1960, 2010.

[9] T. M. Sezgin and R. Davis, “Sketch Interpretation Using Multiscale Models of Temporal Patterns,” *IEEE Journal of Computer Graphics and Applications*, vol. 27, issue 1, pp. 28-37, 2007.

[10] S. Liu and H. Asada, “Adaptive control of deburring robots based on human demonstration data,” *Proc. Int. Conf. on Industrial Electronics, Control, Instrumentation, and Automation*, vol. 2, pp. 946-952, Nov. 1992. Jun. 2007.

[11] U. Ramer, “An iterative procedure for the polygonal approximation of plane curves,” *Computer Graphics and Image Processing*, vol.1, no.3, pp. 244-256, 1972.

[12] D. Douglas, and T. Peucker, “Algorithm for the reduction of the number of points required to represent a digitized line of its caricature,” *The Canadian Cartographer*, vol. 10(2), pp. 112-122, 1973.

[13] J. Hershberer, and J. Snoeyink, “Speeding Up the Douglas-Peucker Line-Simplification Algorithm,” *Proc. 5th Symp. on Data Handling*, pp. 134-143, 1992.

Taeyong Choi
Senior Researcher of the Korea Institute of Machinery & Materials
Research Area: Artificial Intelligent, Robot Control, etc.
E-mail : taeyongc@kimm.re.kr

Hyunmin Do
Senior Researcher of the Korea Institute of Machinery & Materials
Research Area: Mobile robot, Robot Control, etc.
E-mail : hmdo@kimm.re.kr

Chanhun Park
Principal Researcher of the Korea Institute of Machinery & Materials
Research Area: Robot dynamics, Robot Control, etc.
E-mail : chpark@kimm.re.kr

Kwangcho Chung
Principal Researcher of the Korea Institute of Machinery & Materials
Research Area: Nonlinear Control, Robot Control, etc.
E-mail : ckc@kimm.re.kr

Dongil Park
Senior Researcher of the Korea Institute of Machinery & Materials
Research Area: Robot Mechanism, Robot Kinematics, etc.
E-mail : parkstar@kimm.re.kr

Jinho Kyung
Principal Researcher of the Korea Institute of Machinery & Materials
Research Area: Vibration Analysis, Robot Control, etc.
E-mail : jhkyung@kimm.re.kr