

# A Method to Generate Freeform Curves from a Hand-drawn Sketch

Tetsuzo Kuragano and Akira Yamaguchi  
Graduate School of Information Science, Meisei University  
2-590 Nagabuchi, Ome-City, Tokyo, 198-8655, Japan

## ABSTRACT

When designers begin a product design, they create their ideas and expand them. This process is performed on paper, and designers' hand-drawn lines are called sketches. If the designers' hand-drawn sketches can be realized as real curves, it would be effective in shortening the design period. We have developed a method to extract five degree Bézier curves based on a hand-drawn sketch. The basic techniques to detect curves based on a hand-drawn sketch are described. First, light intensity transformation, binarization of the hand-drawn sketch, and feature based erosion and dilation to smooth the edges of the binary sketch image are described. Then, line segment determination using the detected edges is described. Using the determined line segments a five degree Bézier curve generation is described. A curve shape modification algorithm is described. A curve shape modification algorithm is described. A curve shape modification algorithm is described. Examples of five degree fair curvature Bézier curves based on a sketch are given.

**Keywords:** fair curve generation, curvature, hand-drawn sketch, image processing, regression line, principal component analysis, Bézier curve, least-squares method

## 1. INTRODUCTION

Conventional design procedures can easily produce simple shapes. However currently, industrial designers are drawn to aesthetically pleasing freeform shapes because they have great customer appeal, especially in a highly competitive and technically well-developed market, such as that for automobiles and electrical appliances. It takes a long time to design aesthetically pleasing products using conventional procedures.

A solution to this problem is to establish a method to shorten the product design period, especially the period from the first idea generation as it is narrowed down to the final design.

The objective of developing this method is to shorten the product design period. When designers begin a product design, designers create their ideas and expand them. Normally this process is performed on paper, and designers' hand-drawn lines are called sketches. If the designers' rough idea can be realized as a real curve, it will be effective in shortening the design period.

There are many related works for generating curves based on a sketch. Curve generation using a sketch, design language and characteristic lines [1], curve generation based on a hand-drawn sketch [2], curves for a character such as a stuffed animal design based on a hand-drawn sketch [3], 3D shape reconstruction using 3 views [4], and simple polygonal shape

reconstruction based on a hand-drawn sketch [5] have been published.

Curve generation using a hand-drawn sketch and its view points [6], remeshing based mesh smoothing by a sketch [7], shape generation using a volumetric modeling technique [8], and shape generation using 3D scenes [9] have also been published.

In addition to these, mechanical parts such as a piston using a hand-drawn sketch [10], simple parts generated by constructed solid geometry [11], and simple mechanical parts design using digital clay [12] have been published.

There are many related works for generating fair curves. Fair curve generation algorithms related to energy functions have been published. These find the unfair portion of a curve using energy function [13], and apply a low-pass filter to energy function [14].

Fair curve generation algorithms related to curvature control have been published. These make monotone curvature [15], use a clothoidal curve for specifying the curvature [16], and drive the curve based on the specified curvature [17].

In addition to these, fair curve generation by using the second derivative values [18], using an argument of Bézier control edges [19], minimizing positional, the first, the second, and the third derivative values [20] have been published.

In this study, hand-drawn sketches are put into a computer by using a scanner. Then, image processing techniques such as light intensity transformation, edge detection of a hand-drawn sketch, and feature based erosion and dilation to smooth the edge of the binary image are examined.

In addition to these image processing techniques, principal component analysis is introduced to generate line segments using the detected edges. Using the line segments determined, five degree Bézier curves are generated.

Section 2 of this paper describes the freeform curve expression. Section 3 of this paper describes the techniques for image processing such as light intensity transformation, binarization of the sketch image, and feature based erosion and dilation to smooth the edge of the binary image. Section 4 describes the line segment generation based on the detected edges by introducing the principal component analysis. Section 5 describes freeform curve generation using the line segments. The generation of a two degree Bézier curve based on the line segments is described. In addition to this, generation of a five degree Bézier curve elevated from a two degree Bézier curve is described. Section 6 describes a method to generate a fair curve using the specified curvature distribution by introducing the least-squares method. In section 7, examples of Bézier curve generation from a hand-drawn sketch are given.

## 2. FREEFORM CURVE EXPRESSION

A freeform curve is expressed using a five degree Bézier form as in Eq. (1).

$$\mathbf{R}(t) = (1-t+tE)^5 \mathbf{Q}_0, \quad (1)$$

where  $t$  is a parameter which varies between 0 and 1,  $\mathbf{Q}_0, \mathbf{Q}_1, \dots, \mathbf{Q}_5$  are control points, and  $E$  is a shifting operator described by  $E \cdot \mathbf{Q}_i = \mathbf{Q}_{i+1}$  ( $i = 0, 1, 2, \dots, 4$ ).

The first derivative is given by differentiating Eq. (1) with respect to  $t$  and shown as in Eq. (2).

$$\begin{aligned} \dot{\mathbf{R}}(t) &= 5(1-t+tE)^4 (E-1)\mathbf{Q}_0 \\ &= 5(1-t+tE)^4 (\mathbf{Q}_1 - \mathbf{Q}_0) \end{aligned} \quad (2)$$

The second derivative is given by differentiating Eq. (2) with respect to  $t$  as shown in Eq. (3).

$$\begin{aligned} \ddot{\mathbf{R}}(t) &= 5 \cdot 4(1-t+tE)^3 (E-1)^2 \mathbf{Q}_0 \\ &= 5 \cdot 4(1-t+tE)^3 \{(\mathbf{Q}_2 - \mathbf{Q}_1) + (\mathbf{Q}_0 - \mathbf{Q}_1)\} \end{aligned} \quad (3)$$

At both ends of a Bézier curve, the first derivatives become as follows using Eq. (2).

$$\dot{\mathbf{R}}(0) = 5(\mathbf{Q}_1 - \mathbf{Q}_0) \quad (4)$$

$$\dot{\mathbf{R}}(1) = 5(\mathbf{Q}_5 - \mathbf{Q}_4) \quad (5)$$

And the second derivatives become as follows using Eq. (3).

$$\ddot{\mathbf{R}}(0) = 20\{(\mathbf{Q}_2 - \mathbf{Q}_1) + (\mathbf{Q}_0 - \mathbf{Q}_1)\} \quad (6)$$

$$\ddot{\mathbf{R}}(1) = 20\{(\mathbf{Q}_5 - \mathbf{Q}_4) + (\mathbf{Q}_3 - \mathbf{Q}_4)\} \quad (7)$$

Curvature is expressed as in Eq. (8) [21].

$$\kappa(t) = \frac{\ddot{\mathbf{R}}(t) \cdot \mathbf{n}(t)}{(\dot{\mathbf{R}}(t))^2}, \quad (8)$$

where  $\dot{\mathbf{R}}(t)$  is the first derivative of a five degree Bézier curve, and  $\ddot{\mathbf{R}}(t)$  is the second derivative of a five degree Bézier curve.

Curvatures at both ends of a Bézier curve are shown as in Eq. (9) and (10).

$$\kappa(0) = \frac{4(\mathbf{Q}_2 - \mathbf{Q}_1) \cdot \mathbf{n}(0)}{5(\mathbf{Q}_1 - \mathbf{Q}_0)^2} \quad (9)$$

$$\kappa(1) = \frac{4(\mathbf{Q}_3 - \mathbf{Q}_4) \cdot \mathbf{n}(1)}{5(\mathbf{Q}_5 - \mathbf{Q}_4)^2} \quad (10)$$

Equations (1), (4), (5), (6), (7), (9), and (10) are illustrated in Fig.1.

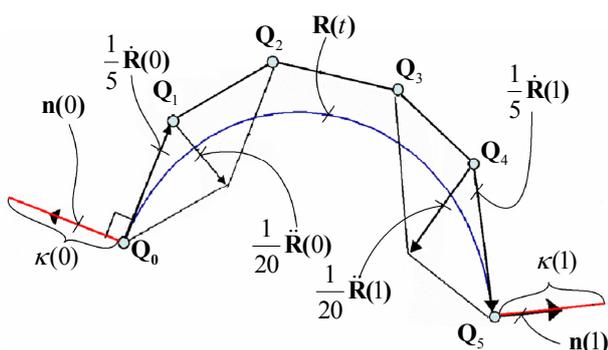


Fig.1 Illustration for a five degree Bézier curve, 1st, 2nd derivatives, and curvatures at both ends

## 3. IMAGE PROCESSING

In this section, the image processing techniques used to detect the edges of a hand-drawn sketch are described. First, light intensity transformation to overcome the gradation given by the designer, binarization, feature based erosion and dilation to smooth the edge of the binary image, and edge detection are described.

### 3.1. Light intensity transformation

Generally, various types of gradation are expressed in the background of a hand-drawn sketch [22] as shown in Fig.2. In such cases, the human eye is able to recognize lines which are drawn in a dark area as well as lines which are drawn in a bright area. However a computer can not recognize lines in a dark area, because of the small difference in light intensity between the lines and the background gradation, which is dark. Therefore, we have developed a method to extract lines which are drawn in a relatively dark area. The original image is transformed into a logarithmic image using Eq. (11).

$$y = a \log x, \quad (11)$$

$a$  is a parameter, which is determined interactively by the designer. The logarithmic image is shown in Fig.3.

In the logarithmic image, the light intensity difference of adjacent image elements in the dark area become very close to the differences in the bright area. In other words, the light intensity of the dark area of the original image is amplified, while that of the bright area of the original image is reduced.



Fig.2 Hand-drawn sketch image Fig.3 Logarithmic image

### 3.2. Binarization of sketch image

A histogram is a graph which shows the frequency of the gray level. When the image signal is analog, this histogram becomes a curve, but when digital, the histogram becomes a bar graph. In case the histogram has two peaks (ridges), the valley part can be a threshold.

But two peaks suggest the possibility of two edges being in existence. In other words, there may exist edges in the relatively dark area and in the relatively bright area. If the valley part shown in Fig.4 is decided as a threshold, we will lose the opportunity to extract the edge in the relatively dark area. In general, designers use various types of gradation in their sketches, so there might be many peaks in the histogram. Therefore, the threshold determination can be performed interactively.

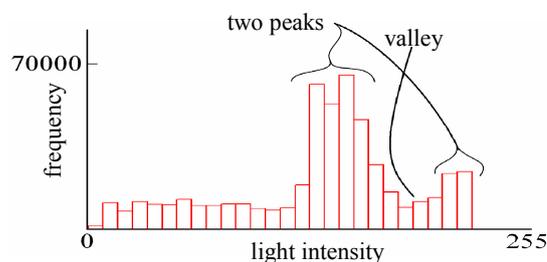


Fig.4 Histogram of hand-drawn sketch image

Binarization should be performed according to the threshold temporarily decided. Then, the candidate points for edges are displayed. This determination process should be done interactively by the designers.

It is difficult to detect edges by using the Laplacian operation, because of the light-intensity gradient of a gray-scale image. Therefore, the entire image has to be binarized for edge detection. A single threshold is not enough in some case to binarize an entire image. Therefore, the entire image is divided into multi regions, and thresholds for each region are established independently. In this case, the entire image is divided into 64 regions. The 64 thresholds for each region are determined and the entire image is then binarized.

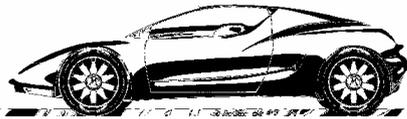


Fig.5 Binarized hand-drawn sketch image

### 3.3. Feature based erosion and dilation to smooth the edge of the binary image

“Erosion and dilation” is a well known method to correct defects in a binary image. They are applied to the binary image as a combination of erosion-dilation or dilation-erosion. Erosion removes granules, and isolated lines and points, dilation fills holes and gaps in the binary image.

The sample image shown in Fig.6a has been created to show the performance difference between “pixel based erosion and dilation” and “feature based erosion and dilation” [23, 24]. Pixel based erosion and dilation is applied 10 times to the image. The result is shown in Fig.6b. Although salt-and-pepper noise [25] is reduced, the image shape becomes very different from the image shown in Fig.6a. This shows that pixel based erosion and dilation is not effective to smooth the edge of the image. To improve this performance, feature based erosion and dilation is applied to the image 10 times. The results are shown in Fig.6c. It can be seen that while keeping the original shape, the salt-and-pepper noise is reduced.

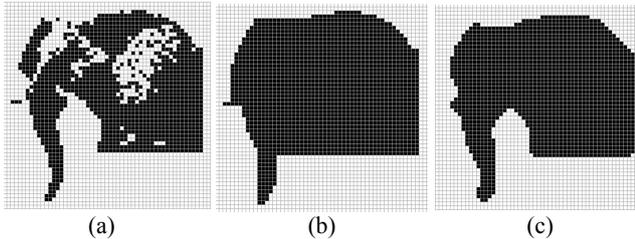


Fig.6 Comparison of pixel based erosion and dilation, and feature based erosion and dilation (a) Sample image (b) Pixel based erosion and dilation applied 10 times (c) Feature based erosion and dilation applied 10 times

### 3.4. Edge detection

The second derivative in the image processing study is called Laplacian. Applying the Laplacian operation to the binary image, the position of the edge in the binary image is detected. This position corresponds to the position of the line in the hand-drawn sketch image. Fig.7a shows the binary image and detected edges. An example of the edge detection is shown in Fig.7b. This shows that the idea described in this section is effective.

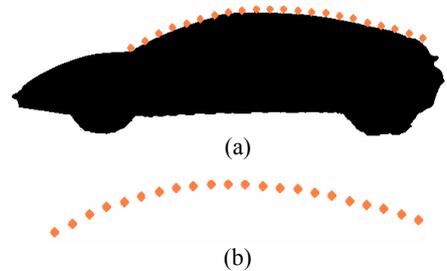


Fig.7 An example of edge detection (a) Binarized image and detected edges (b) Dots are detected edges

The procedure from data input to edge detection is illustrated in Fig.8. The hand-drawn sketch image shown in Fig.8a is put into a computer by using a scanner set at 120 dots per inch.

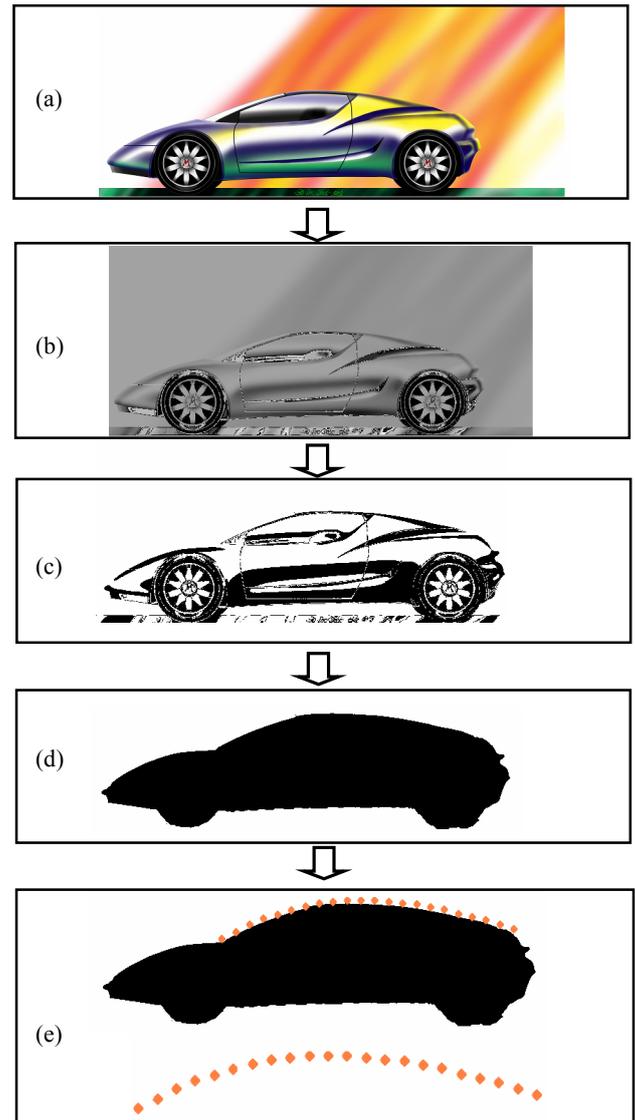


Fig.8 Illustration for the procedure from data input to edge detection (a) Hand-drawn sketch (b) Logarithmic image (c) Binary sketch image (d) Erosion & dilation applied image (e) Edge detected

The logarithmic image is shown in Fig.8b. The binary hand-drawn sketch image is shown in Fig.8c. Feature based erosion and dilation is then applied to the binary sketch image to smooth the edge. The objective of this procedure is to detect the outer edge of the image and to smooth it. Internal image contours are eliminated. The edge smoothed binary sketch image is shown in Fig.8d. By then applying the Laplacian operation, the edges of the smoothed binary sketch image are detected. The detected edges are shown in Fig.8e. Thus, following this procedure, the edges of the sketch image are detected.

#### 4. LINE SEGMENT GENERATION USING THE DETECTED EDGES

A method to determine the regression line for the detected edges is described by showing examples.

When the detected edges are located as shown in Fig.9a, a regression line is generated by using the principal component analysis, the generated regression line becomes a heavily delineated line as shown in Fig.9b. This straight line can not be considered as representative of the edge data.

On the other hand, if lines are drawn that roughly follow the detected edge data points, an image emerges as shown in Fig.9c.

Ordinal numbers are given to each data point. The data points are then placed into 8 groups of 20 points each. This follows the principles of the Sturges rule [26].

The principal component analysis is then applied to each group. A regression line segment representing the edge data for each group is then obtained. In this manner, a line segment for each group is generated.

The intersections of the adjacent line segments are calculated, then concatenated line segments are obtained.

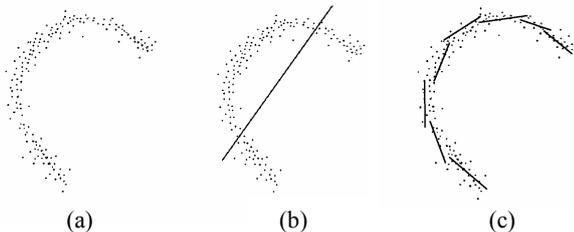


Fig.9 (a) Detected edge data points (b) One line segment representing detected edges (c) 8 line segments representing detected edges

#### 5. BÉZIER CURVE GENERATION BASED ON THE CONCATENATED LINE SEGMENTS

In this section, a method is described to approximate concatenated line segments using Bézier curves. The line segments are shown in Fig.10.

$P_2, P_3, \dots, P_5$  are the intersection of the line segments and  $mp_1, mp_2, \dots, mp_5$  are the midpoints of line segments. Two degree Bézier curves are generated assuming midpoints are placed at both ends of a two degree Bézier curve as control points, and also assuming the intersections of the line segments are considered as central control points. These two degree Bézier curves are heavily delineated as shown in Fig.10. Four two degree Bézier curves are generated based on the concatenated line segments.

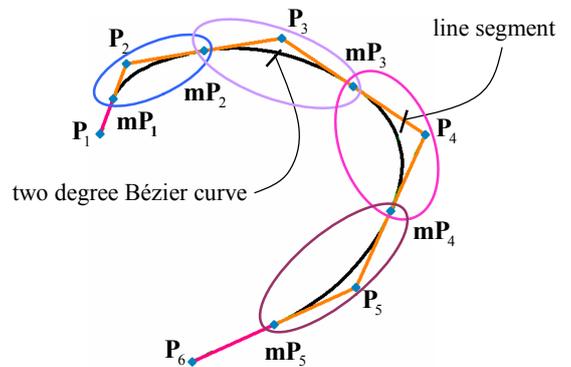


Fig.10 Illustration for Bézier curve generation based on concatenated line segments

The Bézier curves are shown with curvature plots in Fig.11. Curvatures are discontinuous at the joints of the curves as indicated by the arrows. Because both ends of the two degree Bézier curves are at the midpoints of the line segments, the first derivative values are continuous at the curve segment joints as indicated by the arrows in Fig.12.

These two degree Bézier curves are degree elevated to five degrees. The five degree Bézier curves are shown in Fig.13 with their curvature plots. Curvature discontinuities are examined at the joints of the curves as indicated by the arrows.

The first derivative value distribution of degree elevated Bézier curves is shown in Fig.14.

These five degree Bézier curves have exactly the same properties as two degree Bézier curves.

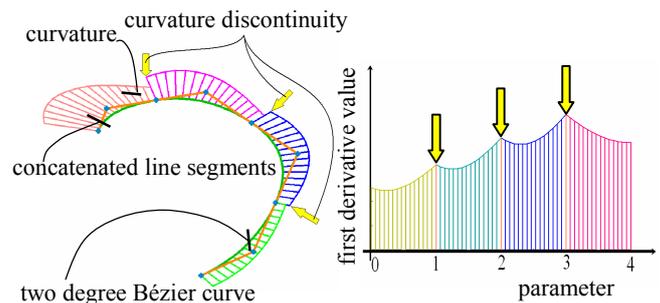


Fig.11 Two degree Bézier curves with curvature plots

Fig.12 The first derivative value distribution of two degree Bézier curves

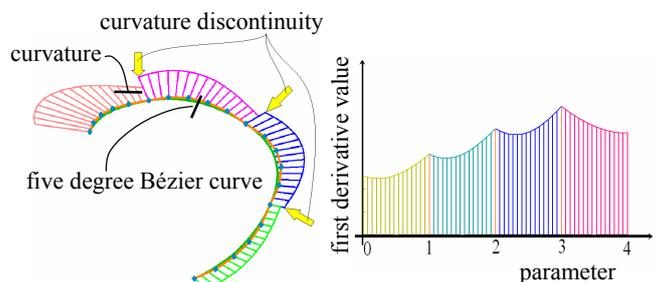


Fig.13 Five degree Bézier curves with curvature plots

Fig.14 The first derivative value distribution of five degree Bézier curves which are elevated from two degrees

## 6. METHOD TO GENERATE A FAIR CURVE BASED ON THE SPECIFIED CURVATURE

In the previous sections, a method to extract Bézier curves based on a hand-drawn sketch which includes the rough idea of the designer was examined. These extracted Bézier curves correspond to the entire external shape or partial external shape of the product to be designed. More specifically, the image processing techniques, such as intensity transformation, binarization, and feature based erosion and dilation are applied to the sketch image for edge detection. Using the detected edges, principal component analysis is applied to generate line segments. Then, using the generated line segments, Bézier curves are generated.

The designer evaluates the quality of the Bézier curves by visual evaluation techniques, such as curvature plots, curvature distribution, and radius of curvature distribution. If the quality of the curve does not coincide with the designer's intention, the designer modifies the shape of the curves by changing the position of the control points. Therefore, it takes a long time to modify the shape of the Bézier curves.

To overcome this problem, a method to generate a five degree Bézier curve based on the specified curvature distribution is described in this section.

To generate a fair five degree Bézier curve, it is assumed no inflection point exists within a curve segment. For easy computation, each control point is redefined as Eq. (12).

$$\mathbf{Q}_n = \mathbf{Q}_{n-1} + a_n \cdot \mathbf{d}_n \quad (12)$$

where

$$\mathbf{d}_n = \frac{\mathbf{Q}_n - \mathbf{Q}_{n-1}}{|\mathbf{Q}_n - \mathbf{Q}_{n-1}|} \quad (n = 1, 2, \dots, 5)$$

and  $a_n$  is the magnitude of the difference of control points  $\mathbf{Q}_n - \mathbf{Q}_{n-1}$ . This redefinition of the control polygon is illustrated in Fig.15.

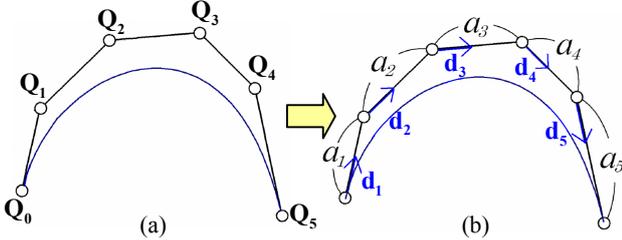


Fig.15 Redefined control points of a five degree Bézier curve (a) Control points of Bézier curve (b) Redefined control polygon

Radius of curvature, which is the reciprocal value of the curvature, can also be considered to modify the shape of the curve. But in a case where a curve shape is very close to a straight line, the radius of curvature becomes infinity. Therefore, curvature should be dealt with instead of the radius of curvature.

The concept of curve shape modification based on the specified curvature is illustrated in Fig.16.

The differences  $\delta_i (i = 0, 1, 2, \dots, m)$  between the Bézier curve curvature  $\kappa_i$  and specified curvature  $\hat{\kappa}_i$  are shown in Eq. (13).  $S(a_1, a_2, \dots, a_5)$  which is the sum of the squared deviations for all specified curvatures in Eq. (14) should be minimized by introducing the least-squares method.

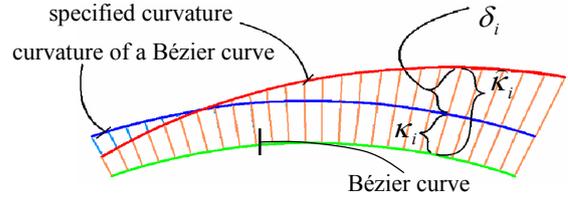


Fig.16 Concept of curve shape modification algorithm based on the specified curvature

$$\delta_i = \kappa_i(a_1, a_2, \dots, a_5) - \hat{\kappa}_i \quad (13)$$

$$S(a_1, a_2, \dots, a_5) = \sum_{i=0}^m [\kappa_i(a_1, a_2, \dots, a_5) - \hat{\kappa}_i]^2 \quad (14)$$

$m$  is the number of specified curvatures.

The curvature expression is non-linear. Therefore, by Taylor's theorem, Eq. (14) is linearized as Eq. (15).

$$S(a_1 + \Delta a_1, a_2 + \Delta a_2, \dots, a_5 + \Delta a_5) = \sum_{i=0}^m \left[ \kappa_i(a_1, a_2, \dots, a_5) + \frac{\partial \kappa_i}{\partial a_1} \Delta a_1 + \dots + \frac{\partial \kappa_i}{\partial a_5} \Delta a_5 - \hat{\kappa}_i \right]^2 \quad (15)$$

$m$  is the number of specified curvatures.

To minimize  $S(a_1 + \Delta a_1, a_2 + \Delta a_2, \dots, a_5 + \Delta a_5)$  is achieved by equating to zero all partial derivatives of  $S(a_1 + \Delta a_1, a_2 + \Delta a_2, \dots, a_5 + \Delta a_5)$  with respect to  $\Delta a_r (r = 1, 2, \dots, 5)$  as Eq. (16).

$$\frac{\partial S}{\partial \Delta a_r} = 0 \quad (r = 1, 2, \dots, 5) \quad (16)$$

Using these simultaneous linear equations,  $\Delta a_r (r = 1, 2, \dots, 5)$  are calculated. Then,  $a_r (r = 1, 2, \dots, 5)$  are determined.

To show this method is effective, an example is given. Five degree Bézier curves with curvature plots are shown in Fig.17 and their curvature distribution is shown in Fig.18. Curvatures are discontinuous at the joints of the curve segments as indicated by the arrows. For this curvature distribution, a new curvature distribution is specified as shown by the dotted line in Fig.18. The curve shape modification algorithm is applied to these five degree Bézier curves. Then, five degree Bézier curves with specified curvature plots as shown in Fig.19 are generated. A curvature distribution of a generated five degree Bézier curve of 3 segments and the specified curvature shown in Fig.18 are shown in Fig.20. The curvature distribution of shape modified curve and specified curvature distribution look the same. This means that the curve shape modification algorithm mentioned above is effective.

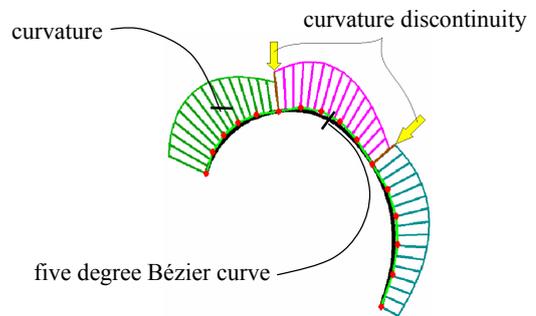


Fig.17 Five degree Bézier curve with curvature plots

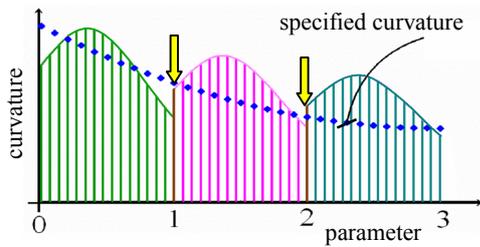


Fig.18 Curvature distribution and specified curvature

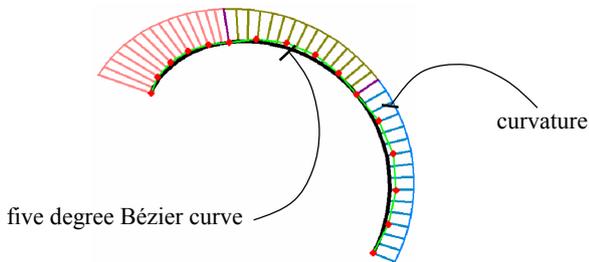


Fig.19 Generated Bézier curve of 3 segments with specified curvature

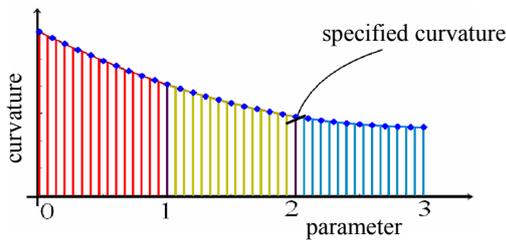


Fig.20 Curvature distribution of generated Bézier curve of 3 segments and the specified curvature shown in Fig. 18

## 7. EXAMPLES OF BÉZIER CURVE GENERATION BASED ON A HAND-DRAWN SKETCH

Using the techniques mentioned in the previous sections, a five degree Bézier curve generation process is described according to the following steps.

- 1) Fig.21 shows a hand-drawn sketch of the side-view of an automobile. This sketch is put into the computer via a scanner. A gradated background is not shown in this hand-drawn sketch. Therefore, transformation to a logarithmic image is unnecessary. Then, the hand-drawn sketch is binarized. The edge of the binary sketch image is smoothed by applying feature based erosion and dilation for edge detection.
- 2) The detected edges indicated by dots are shown with the hand-drawn sketch as shown in Fig.22.
- 3) Using the detected edges, regression line segments are determined. These line segments are shown in Fig.23.
- 4) Using the determined line segments, five degree Bézier curves are generated. These curves with curvature plots are shown in Fig.24. Curvatures are discontinuous at the joints of the curve segments as indicated by the arrows.
- 5) Curvature continuity can be realized using Eq. (9) and (10) at both ends (indicated by big dots in Fig.25) of the Bézier curve segments. Curvature continuity is examined by the arrows indicated in Fig.25.

In addition to this, the low pass filtering technique [27, 28] is applied to this curvature. Then the curvature smoothed

five degree Bézier curves are generated. These are shown in Fig.25.

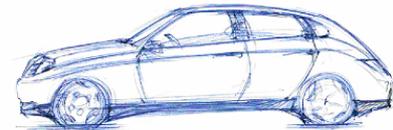


Fig.21 Hand-drawn sketch

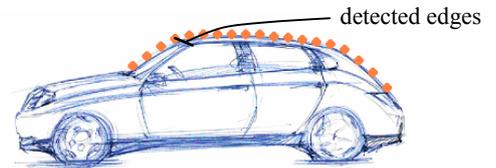


Fig. 22 Detected edges

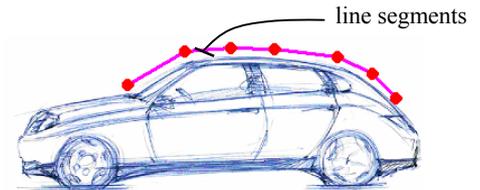


Fig.23 Concatenated line segments

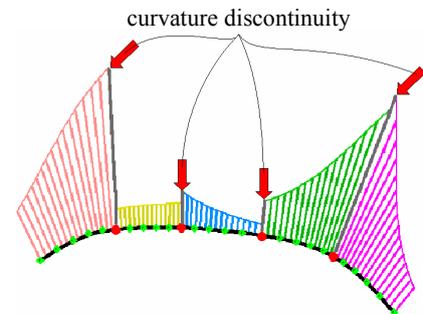


Fig.24 Generated five degree Bézier curves with curvature plots

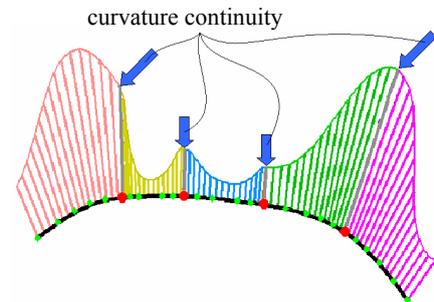


Fig.25 Curvature smoothed five degree Bézier curves and their curvature plots

## 8. CONCLUSIONS

The objective of this study is to shorten the product design period by extracting a freeform curve from a hand-drawn sketch.

A freeform curve is expressed as a five degree Bézier form. The first derivative, the second derivative, and curvatures at both ends of a Bézier curve are illustrated.

Edge smoothing of the binary sketch image by feature based erosion and dilation, and line segment generation using the detected edges are described.

In addition to these, Bézier curve generation based on the concatenated line segments is described.

Then, a method to generate a fair curve, which is aesthetically pleasing, based on the specified curvature distribution is described.

Examples of Bézier curve generation based on a hand-drawn sketch are given according to the generation steps.

We have suggested an edge smoothing method of the binary image by feature based erosion and dilation, and five degree Bézier curve generation with curvature continuity based on the line segments.

We have also suggested a method to fair the shape of five degree Bézier curves by introducing the curve shape modification algorithm based on the specified curvature distribution.

As a future issue, a method to specify curvature will be studied.

## 9. REFERENCES

- [1] H. Aoyama, Y. Urabe, M. Ohta, and T. Kusunoki "Aesthetic design system based on sketch, design language, and characteristic lines", *CIRP Journal of Manufacturing systems*, vol 32, (2003), pp. 101-106.
- [2] S. Saga, "A freehand interface for computer aided drawing systems based on the fuzzy spline curve identifier", *IEEE*, (1995), pp. 2754-2759.
- [3] T. Igarashi, S. Matsuoka, H. Tanaka, "Teddy: A sketching interface for 3D freeform design", *SIGGRAPH 99*, (Aug,99), pp. 409-416.
- [4] J. Mitani, H. Suzuki, F. Kimura, "3D Sketch: Sketch-based model reconstruction and rendering", *7th IFIP WG5.2 International Workshop on Geometric Modeling (GEO-7)*, (2000), pp. 85-112.
- [5] S. Sugishita, K. Kondo, H. Sato, S. Shimada, and F. Kimura, "Interactive freehand sketch interpreter for geometric modeling", *Symbiosis of Human and Artifact*, (1995), pp. 561-566.
- [6] K. Matsuda, S. Sugishita, K. Kondo, H. Sato, and S. Shimada, "Freehand sketch system for 3D geometric modeling", *IEEE*, (1997), pp.55-62.
- [7] C.C.L. Wang, Y. Wang, M.M.F. Yuen, "Remeshing based mesh smoothing by 2Dsketches input. *Proceedings of DETC'02 ASME 2002 Design Engineering Technical conference and Computer and Information in Engineering Conference*", (Oct,2002), pp.579-587.
- [8] T.A. Galyean, J.F. Hughes, "Sculpting : An interactive volumetric modeling technique", *Proc. ACM SIGGRAPH '91*, Vol 25, Num 4, (Jul,91), pp. 267-273.
- [9] R.C. Zeleznik, K.P. Herndon, J.F. Hughes, "SKETCH : An interface for sketching 3Dscenes" *Computer graphics Proceedings*, (1996), pp. 163-170.
- [10] Y. Zeng Y, A. Pardasani, H. Antunes, Z.Li, J. Dickinson, V. Gupta, D. Baulier, "Representation and interpretation of sketches in mechanical design: experimental and theoretical approaches", *Proceedings of DETC'03 ASME 2003 Design Engineering Technical conferences and Computers and Information in Engineering conference*, (Sep,2003), pp. 613-624.
- [11] A. Shesh, and B. Chen, "SMARTPAPER : An interactive and user friendly sketching system", *EUROGRAPHICS 2004*, vol 23 (2004), pp. 301-310.
- [12] E. Schweikardt, M.D. Gross, "Digital clay : deriving digital models from free hand sketches", *Automation in Construction*, 9 (2000), pp. 107-115.
- [13] C. Zhang, P. Zhang, F.(F). Cheng, "Fairing spline curves and surface by minimizing energy", *Computer Aided Design*, 33, (2001), pp.913-923.
- [14] X. Yang, G. Wang, "Planar point set fairing and fitting by arc splines", *Computer Aided Design*, 33 (2001), pp. 35-43.
- [15] W.H. Frey, D.A. Field, "Designing Bézier conic segments with monotone curvature", *Computer Aided Geometric Design*, 17, (2000), pp. 457-483.
- [16] M. Kuroda, M. Higashi, T. Saitoh, Y. Watanabe, T. Kuragano, "Interpolating curve with B-spline curvature function", *Mathematical Methods for Curves and Surfaces II*, Vanderbilt University Press, Nashville, TN, (1998), pp. 303-310.
- [17] W. Li, S. Xu, J. Zheng, G. Zhao, "Target curvature driven fairing algorithm for planar cubic B-spline curves", *Computer Aided Geometric Design*, 21, (2004), pp. 499-513.
- [18] H. Nowacki, D. Liu, X. Lu, "Fairing Bézier curves with constraints", *Computer Aided Geometric Design*, 7 (1990), pp. 43-55.
- [19] Y. Mineur, T. Lichah, J.M. Castelain, H. Giaume, "A shape controled fitting method for Bézier curves", *Computer Aided Geometric Design*, 15, (1998), pp. 879-891.
- [20] L. Fang, D.C. Gossard, "Multidimensional curve fitting to unorganized data points by nonlinear minimization", *Computer Aided Design*, 27,1 (Jan,1995), pp. 48-58.
- [21] M. Hosaka, "Modeling of curves and surfaces in CAD/CAM", Springer-Verlag, (1992), pp. 61.
- [22] J. Unger, "Rendering in mixed media", (Watson-Guptill Publications, (1985).
- [23] J.C. Russ J.C, *The image processing handbook* forth edition, CRC Press, Florida , (2002), pp. 123-124.
- [24] P. Soille, *Morphological image analysis*, Springer-Verlag, Berlin, (1999).
- [25] R.C. Gonzalez, and R.E. Woods, *Digital image processing* second edition, prentice Hall, Upper Saddle River, New Jersey, 2002, pp. 123-124.
- [26] E.A. Avallone, P. Barry, G. Bonn, W. G. Bowman, "McGraw-Hill Dictionary of Scientific and technical terms Sixth Edition", McGraw-Hill, (2002), pp.2054.
- [27] T. Kuragano, Y. Matsumura, A. Yamaguchi, S. Furukawa, "Method to measure curve shape similarities and to generate similar curves", *The 8th World Multi-Conference on Systemics, Cybernetics and Informatics*, (2004), pp. 137-143.
- [28] T. Kuragano, A. Yamaguchi, S. Furukawa, "A method to measure foot print similarity for gait analysis", *IEEE CIMCA '05*, (Nov, 2005).