

A FAIR CURVE GENERATION USING A HAND-DRAWN SKETCH FOR COMPUTER AIDED AESTHETIC DESIGN

Akira Yamaguchi¹ and Tetsuzo Kuragano¹

¹Meisei University, Tokyo, Japan

ABSTRACT

Image processing techniques are used to detect the edges of a hand-drawn sketch. Generally, various types of gradation are expressed in the background of a hand-drawn sketch. Therefore, the original image is transformed into a logarithmic image. The entire image has to be binarized for edge detection. To smooth the edge of the binary image, feature based erosion and dilation is applied. The Laplacian operation is applied to the binary image to detect the edge in the binary image. This edge position corresponds to the position of the line in the hand-drawn sketch image. Line segments which represent the detected edges are generated. Using line segments, a NURBS curve is generated. Using the position and gradient on the line segments, a NURBS curve is generated. The shape of a NURBS curve extracted from the sketch image is examined by the designer visually. The shape of this NURBS curve generally coincides with the intention of the designer, but does not coincide precisely. Therefore, a curve shape modification method based on the specified radius of curvature is applied. The sum of the squared differences between the radius of curvature of a curve and the specified radius of curvature of an entire curve is linearized by Taylor's theorem, then minimized.

Keywords: Hand-drawn sketch, image processing, regression line, NURBS curve, curvature, radius of curvature, curve shape modification

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. Therefore, 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 on the sketch can be realized as a real curve, it will be effective in shortening the design period. As a drawing technique, there are two canons. One is perspective projection. The other is orthogonal projection. Generally, it is not easy to estimate the perspective transformation matrix from a sketch drawn by perspective projection. Therefore, the sketch in this study is limited to orthogonal projections such as front view, side view and top view.

First, 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 sketch image are examined.

In addition to these image processing techniques, principal component analysis is introduced to generate regression line segments using the detected edges. Using the line segments determined, a quintic NURBS curve is generated. A NURBS curve, which is commonly used in the area of CAD·CAM and Computer Graphics, is used as an expression of a freeform curve. A quadratic NURBS curve is used as an expression of a quadratic curve using its weights. In this study, a quadratic curve is not used to express the shape of a curve. Therefore, the weights of the NURBS curve are not used. A

cubic NURBS curve is widely used also, but in this study, radius of curvature over the multi segments of the NURBS curve is modified based on the specified radius of curvature. So, a smooth radius of curvature continuity is needed. Therefore, a quintic NURBS curve is used in this study.

The position of the midpoints and the gradient of the line segments are given to the NURBS curve equation and first derivative equation of the NURBS curve respectively. Then, a NURBS curve is generated. Afterwards, if necessary, the shape of this NURBS curve is modified according to the specified radius of curvature distribution.

There are many related works for generating curves based on a sketch. Curve generation by tracing a hand-drawn sketch is available as a function of a commercial computer aided aesthetic design system [1]. Curve generation using a sketch, design language and characteristic lines [2], curve generation based on a hand-drawn sketch [3], curves for a character such as a stuffed animal design based on a hand-drawn sketch [4], 3D shape reconstruction using hand-drawn lines [5], and simple polygonal shape reconstruction based on a hand-drawn sketch [6] have been published. Curve generation using a hand-drawn sketch and its view points [7], remeshing based mesh smoothing by a sketch [8], shape generation using a volumetric modeling technique [9], and shape generation using 3D scenes [10] have also been published.

In addition to these, mechanical parts such as a piston using a hand-drawn sketch [11], simple parts generated by constructed solid geometry [12], and simple mechanical parts design using digital clay [13] 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 [14], and apply a low-pass filter to energy function [15]. Fair curve generation algorithms related to curvature control have been published. These make monotone curvature [16], use a clothoidal curve for specifying the curvature [17], and modify the curve based on the specified curvature [18]. In addition to these, fair curve generation by using the second derivative values [19], using an argument of Bézier control edges [20], minimizing positional, the first, the second, and the third derivative values [21] have been published.

Section 2 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 3 describes the line segment generation based on the detected edges by introducing the principal component analysis. Section 4 of this paper describes NURBS curve expression and NURBS curve shape modification based on the specified radius of curvature. Section 5 describes NURBS curve generation using the line segments. In section 6, examples of NURBS curve generation based on a hand-drawn sketch according to the generation process are given.

2 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.

2.1 Light intensity transformation

Generally, various types of gradation are expressed in the background of a hand-drawn sketch [22] as shown in Figure 1. 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.(1).

$$y = a \log x, \tag{1}$$

where a is a factor, which is determined interactively by the designer. The logarithmic image is shown in Figure 2. 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.

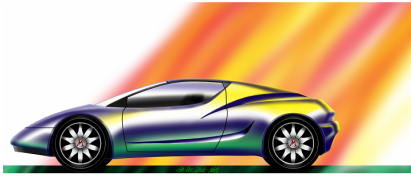


Figure 1. Hand-drawn sketch image

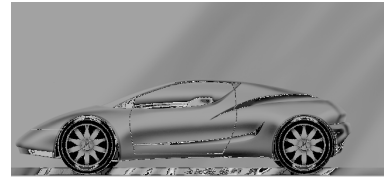


Figure 2. Logarithmic image

2.2 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 more 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 Figure 3 is decided as a threshold, we will lose the opportunity to extract the edges in the relatively dark area and in the relatively bright 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. 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.

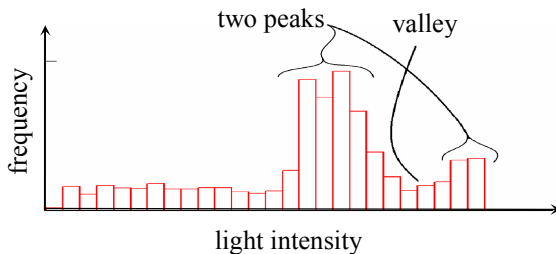


Figure 3. Histogram of hand-drawn sketch image

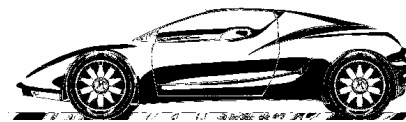
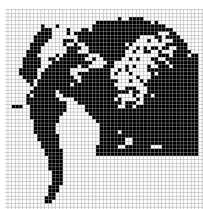


Figure 4. Binarized hand-drawn sketch image

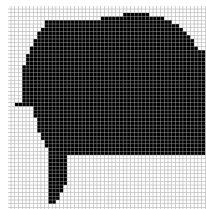
A single threshold is not enough in some cases to binarize an entire image. Therefore, the entire image is divided into multi regions, and thresholds for each region are established independently. In this case, the entire image is divided into 64 regions. The 64 thresholds for each region are determined and the entire image is then binarized.

2.3 Feature based erosion and dilation to smooth the edge of the binary image

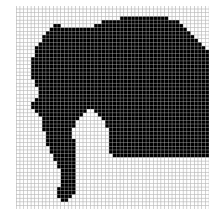
“Erosion and dilation” is a well-known method to correct defects in a binary image. They are applied to the binary image as a combination of erosion-dilation or dilation-erosion. Erosion removes granules, and isolated lines and points, dilation fills holes and gaps in the binary image. The sample image shown in Figure 5(a) has been created to show the performance difference between “pixel based erosion and dilation” and “feature based erosion and dilation” [23, 24].



(a) sample image



(b) pixel based erosion and dilation applied 10 times



(c) feature based erosion and dilation applied 10 times

Figure 5. Comparison of pixel based erosion and dilation, and feature based erosion and dilation

Pixel based erosion and dilation is applied 10 times to the image. The result is shown in Figure 5(b). Although salt-and-pepper noise [25] is reduced, the image shape becomes very different from the image shown in Figure 5(a). 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 examined. "Feature based erosion and dilation" is used for the purpose of smoothing the edge of the binary image. The binary image is traditionally set to one for black, and to zero for white. The sum of the pixel values of the 8 neighborhoods of the central pixel is called the neighborhood coefficient [23]. It is used to determine the central pixel value, which is zero or one. A neighborhood coefficient varies from zero to eight. It is known by our experiments that a neighborhood coefficient of over 5 does not work. In addition to this, in case the neighborhood coefficient is zero, this process is exactly the same as "pixel based erosion and dilation". Therefore, a neighborhood coefficient is determined interactively using the value from one to four, according to the visual inspection of the binary image.

For the erosion process, if the sum of the neighborhoods is bigger than the neighborhood coefficient, the central pixel value is set to zero, and if the sum of the neighborhoods is smaller than the neighborhood coefficient, the central pixel value is set to one.

For the dilation process, if the sum of the neighborhoods is bigger than the neighborhood coefficient, the central pixel value is set to one, and if the sum of the neighborhood is smaller than the neighborhood coefficient, the central pixel value is set to zero.

As mentioned above, for the process of erosion and dilation, a method to decide the central pixel value, according to the sum of 8 neighborhoods based on the neighborhood coefficients is called "feature based erosion and dilation" [23] in this paper.

Feature based erosion and dilation is applied to the image 10 times. The result is shown in Figure 5(c). It can be seen that while keeping the original shape, the salt-and-pepper noise is reduced.

2.4 Edge detection

The first derivative for a digital image is performed by difference. A Laplacian in the area of image processing is the second derivatives, so a Laplacian operation is performed by the difference of the first derivative. Therefore, a Laplacian operator is expressed by the difference of the difference. As a Laplacian operator, there exist 3×3 , 5×5 , 7×7 , 13×13 and so on. Our purpose of using a Laplacian operator is to detect the edge of the binary image. For this reason, a Laplacian operator is decided as 3×3 which is 8 neighborhoods. If a Laplacian operator is applied to the binary image, the Laplacian image has a positive and negative value, between these two values, the Laplacian value becomes zero. This is called zero crossing. The position for zero crossing is detected as the edge of the binary image. Assuming $f(i, j)$ is the pixel value of position i, j of binary image, 8 neighborhoods are expressed as shown in Figure 6. Since the Laplacian operator is the second derivative, which is defined by the difference of the difference, the operator is expressed as shown in Figure 7.

$f(i-1, j-1)$	$f(i-1, j)$	$f(i-1, j+1)$
$f(i, j-1)$	$f(i, j)$	$f(i, j+1)$
$f(i+1, j-1)$	$f(i+1, j)$	$f(i+1, j+1)$

Figure 6. Pixel value of 8 neighborhoods

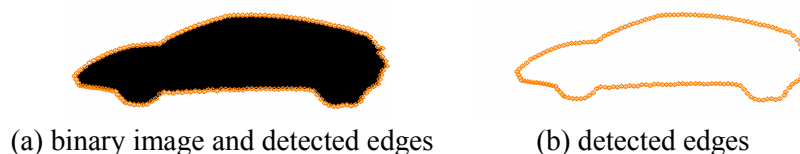
1	1	1
1	-8	1
1	1	1

Figure 7. Laplacian operator for 8 neighborhoods

The pixel value $g(i, j)$ of the Laplacian image corresponding to $f(i, j)$ of the binary image is expressed by Eq.(2) using Figure 6 and Figure 7.

$$g(i, j) = f(i-1, j-1) + f(i-1, j) + f(i-1, j+1) + f(i, j-1) + f(i, j+1) + f(i+1, j-1) + f(i+1, j) + f(i+1, j+1) - 8f(i, j) \quad (2)$$

Figure 8(a) shows the binary image and detected edges. An example of detected edges is shown in Figure 8(b). This shows that the idea described in this section is effective.

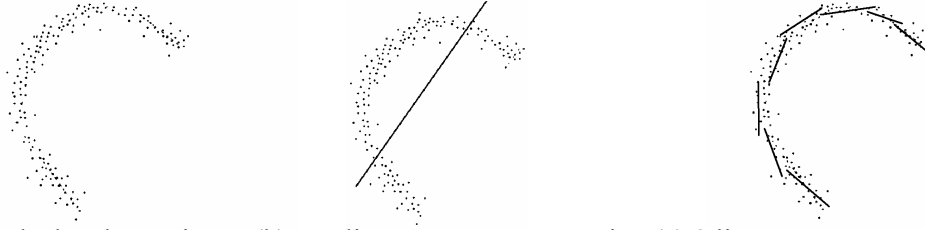


(a) binary image and detected edges (b) detected edges

Figure 8. An example of edge detection

3 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 160 detected edges are located as shown in Figure 9(a), a regression line is generated by using the principal component analysis, the generated regression line becomes a heavily delineated line as shown in Figure 9(b). 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 Figure 9(c).



(a) detected edge data points (160 points) (b) one line segment representing detected edges (c) 8 line segments representing detected edges

Figure 9. Detected edge data and their representative lines

For the principal component analysis, we have to place the data points into the desirable number of groups containing the desirable number of data points. But we have no rule for grouping. Therefore, although the principal component analysis is not the statistical analysis, we follow the principle of Sturges rule [26]. The data points are then placed into 8 groups of 20 points each in this case. 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, number of groups and number of detected edge data in all groups are determined by following the principle of Sturges rule. Then, regression line segments for all groups are generated using the detected edges from the hand-drawn sketch.

4 NURBS CURVE EXPRESSION AND MODIFICATION BASED ON THE SPECIFIED RADIUS OF CURVATURE

A quintic NURBS curve consists of $n-5$ segments ($n \geq 6$) is composed of n control points such as q_0, q_1, \dots, q_{n-1} and n weights such as $\omega_0, \omega_1, \dots, \omega_{n-1}$ in Eq.(3).

$$R(t) = \frac{\sum_{i=0}^{n-1} N_{i,6}(t) \cdot \omega_i \cdot q_i}{\sum_{i=0}^{n-1} N_{i,6}(t) \cdot \omega_i}, \quad (3)$$

where $N_{i,6}(t)$ ($i = 0, 1, \dots, n-1$) are NURBS basis functions.

These functions are recursively defined by knot sequence t_0, t_1, \dots, t_{n+5} as in Eq.(4).

$$N_{i,1}(t) = \begin{cases} 1 & (t_i \leq t < t_{i+1}) \\ 0 & \text{otherwise} \end{cases}, \quad (4)$$

$$N_{i,M}(t) = \frac{t - t_i}{t_{i+M-1} - t_i} N_{i,M-1}(t) + \frac{t_{i+M} - t}{t_{i+M} - t_{i+1}} N_{i+1,M-1}(t)$$

where $i = 0, 1, \dots, n-1$ and $M = 2, 3, \dots, 6$.

The basis functions are defined by the de Boor-Cox [27] recursion formulas. If the knot vector contains a sufficient number of repeated knot values, then a division of the form $N_{i,M-1}(t)/(t_{i+M-1} - t_i) = 0/0$ (for some i) may be encountered during the execution of the recursion. Whenever this occurs, it is assumed that $0/0 = 0$ [28].

A quintic NURBS curve with knot vector $\{-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6\}$ is expressed as in Eq.(5).

$$R(t) = \frac{1}{120} \left\{ (1-t)^5 q_0 + (5t^5 - 20t^4 + 20t^3 + 20t^2 - 50t + 26) q_1 + (-10t^5 + 30t^4 - 60t^2 + 66) q_2 \right. \\ \left. + (10t^5 - 20t^4 - 20t^3 + 20t^2 + 50t + 26) q_3 + (-5t^5 + 5t^4 + 10t^3 + 10t^2 + 5t + 1) q_4 + t^5 q_5 \right\} \quad (5)$$

The first derivative of a quintic NURBS curve is expressed as in Eq.(6).

$$\frac{d\mathbf{R}(t)}{dt} = \frac{1}{120} \{ -5(1-t)^4 \mathbf{q}_0 + (25t^4 - 80t^3 + 60t^2 + 40t - 50) \mathbf{q}_1 + (-50t^4 + 120t^3 - 120t) \mathbf{q}_2 + (50t^4 - 80t^3 - 60t^2 + 40t + 50) \mathbf{q}_3 + (-25t^4 + 20t^3 + 30t^2 + 20t + 5) \mathbf{q}_4 + 5t^4 \mathbf{q}_5 \} \quad (6)$$

Curvature vector is expressed by Eq.(7).

$$\boldsymbol{\kappa}(t) = \frac{(\dot{\mathbf{R}}(t) \times \ddot{\mathbf{R}}(t)) \times \dot{\mathbf{R}}(t)}{(\dot{\mathbf{R}}(t))^4}, \quad (7)$$

where $\dot{\mathbf{R}}(t)$ is the first derivative of a NURBS curve, and $\ddot{\mathbf{R}}(t)$ is the second derivative of a NURBS curve. Curvature is the magnitude of the curvature vector, therefore curvature is expressed as in Eq.(8).

$$\kappa(t) = |\boldsymbol{\kappa}(t)| \quad (8)$$

By definition, the curvature of a plane curve is non-negative. However, in many cases it is useful to ascribe a sign to the curve [29]. The choosing of the sign is commonly connected with the tangent rotation (in moving along the curve in the direction of the increasing parameter): The curvature of the curve is positive when its tangent rotates counter-clockwise, the curvature of the curve is negative when its tangent rotates clockwise.

Radius of curvature is the reciprocal number of curvature, therefore, radius of curvature is expressed as in Eq.(9).

$$\rho(t) = \frac{1}{\kappa(t)} \quad (9)$$

If the radius of curvature to the perimeter is linear, curvature distribution will be parabolic. On the contrary, if the curvature to the perimeter is linear, radius of curvature distribution will be parabolic.

Radius of curvature is suitable, because it corresponds to our visual recognition of the shape of the curve. In a case where curve shape is very close to a straight line, the radius of curvature becomes infinity. And also, at the point of inflexion, curvature value becomes zero. Therefore, the radius of curvature value becomes infinite. For these reasons, the radius of curvature value is converted to curvature value for computation.

The concept of radius of curvature specification and NURBS curve shape modification based on the specified radius of curvature is shown in Figure 10. A NURBS curve and its radius of curvature plots are shown in Figure 10(a). The modification of the shape of the NURBS curve shown in Figure 10(a) to that shown in Figure 10(b) is examined.

The radius of curvature plots shown in Figure 10(a) are drawn perpendicular to the curve using straight lines. The length of the line is proportional to the radius of curvature at that spot on the curve. However, the straight lines are not parallel to each other and the beginning points of the individual straight lines are different. So, the curve with radius of curvature display is suitable to examine the variation of radius of curvature as a whole. However, it is not suitable to examine the length of the straight line and variation of radius of curvature visually.

Therefore, considering the parameter of the NURBS curve is different from the perimeter of the curve, the perimeter of the NURBS curve is placed on the horizontal axis as a straight line, and the radius of curvature is placed on the vertical axis as shown in Figure 10(c). Then, the radius of curvature distribution to the perimeter is drawn. After this, the specified radius of curvature is superimposed on the current radius of curvature distribution.

As an example, the linear algebraic function as a specified radius of curvature specification is shown in Figure 10(c). Coefficients of this linear function are calculated by introducing the least-squares method using the current radius of curvature distribution.

The i th of radius of curvature distribution of a perimetrically represented NURBS curve is denoted ρ_i , the specified radius of curvature at the same spot is denoted $\hat{\rho}_i$, the difference δ_i is shown by Eq.(10), and is illustrated in Figure 10(c).

$$\delta_i = \rho_i(q_1^x, \dots, q_{n-2}^x, q_1^y, \dots, q_{n-2}^y) - \hat{\rho}_i \quad (10)$$

where $i=0,1,2,\dots,m-1$, m is the number of specified radius of curvature, and n is the number of NURBS curve segments plus 5, which is the degree of the curve.

$S(q_1^x, \dots, q_{n-2}^x, q_1^y, \dots, q_{n-2}^y)$ which is the sum of the squared differences for all specified radius of curvatures in Eq.(11) should be minimized by introducing the least-squares method. The radius of curvature expression is non-linear. Therefore, by Taylor's theorem, Eq.(11) is linearized as in Eq.(12).

$$S(q_1^x, \dots, q_{n-2}^x, q_1^y, \dots, q_{n-2}^y) = \sum_{i=0}^{m-1} [\rho_i(q_1^x, \dots, q_{n-2}^x, q_1^y, \dots, q_{n-2}^y) - \hat{\rho}_i]^2 \quad (11)$$

$$S(q_1^x + \Delta q_1^x, \dots, q_{n-2}^x + \Delta q_{n-2}^x, q_1^y + \Delta q_1^y, \dots, q_{n-2}^y + \Delta q_{n-2}^y) = \sum_{i=0}^{m-1} \left[\rho_i(q_1^x, \dots, q_{n-2}^x, q_1^y, \dots, q_{n-2}^y) + \frac{\partial \rho_i}{\partial q_1^x} \Delta q_1^x + \dots + \frac{\partial \rho_i}{\partial q_{n-2}^x} \Delta q_{n-2}^x + \frac{\partial \rho_i}{\partial q_1^y} \Delta q_1^y + \dots + \frac{\partial \rho_i}{\partial q_{n-2}^y} \Delta q_{n-2}^y - \hat{\rho}_i \right]^2 \quad (12)$$

To minimize Eq.(12) is achieved by equating to zero all the partial derivatives of $S(q_1^x + \Delta q_1^x, \dots, q_{n-2}^x + \Delta q_{n-2}^x, q_1^y + \Delta q_1^y, \dots, q_{n-2}^y + \Delta q_{n-2}^y)$ with respect to $\Delta q_r^x + \Delta q_r^y$, and Δq_r^y ($r=1, 2, \dots, n-2$) as in Eq.(13).

$$\left. \begin{aligned} \frac{\partial S}{\partial \Delta q_r^x} &= 0 \quad (r=1, 2, \dots, n-2) \\ \frac{\partial S}{\partial \Delta q_r^y} &= 0 \quad (r=1, 2, \dots, n-2) \end{aligned} \right\} \quad (13)$$

Using these simultaneous linear equations, Δq_r^x and Δq_r^y ($r=1, 2, \dots, n-2$) are calculated. Then, q_r^x, q_r^y are determined.

A reverse computation technique is applied to solve this problem. This kind of study on the radius of curvature, or the curvature to realize a fair curve is called a constrained non-linear minimization problem [30]. For computation, ρ_i and $\hat{\rho}_i$ are calculated based on the perimeter. Then, the perimeter used is converted to the parameter to calculate the position of the control points of the NURBS curve. Next, a NURBS curve is generated. The total length of the curve, which is the perimeter, is calculated and rescaled as 1. Repeating these operations, the positions of the control points of the NURBS curve are determined while δ_i ($i=0, 1, \dots, m-1$) are minimized for the entire perimeter.

Using the above mentioned method of a linear algebraic function to specify the radius of curvature shown in Figure 10(c), radius of curvature distribution is changed to the one shown in Figure 10(d), while modifying the shape of the curve. The dotted line shown in Figure 10(d) is a linear algebraic function specifying the radius of curvature distribution shown in Figure 10(c). It is visually recognized that the radius of curvature distribution of the shape modified curve shown in Figure 10(d) matches to the specified radius of curvature.

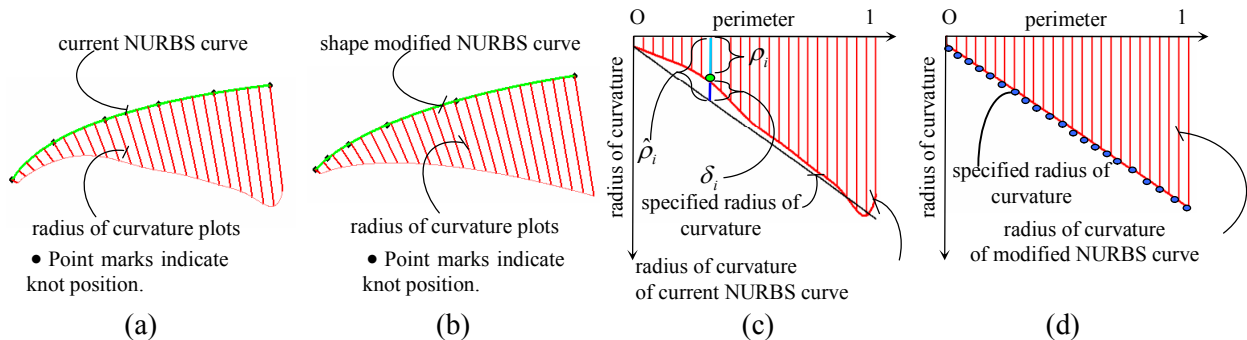


Figure 10. Concept of radius of curvature specification and NURBS curve shape modification based on the specified radius of curvature

- (a) : current NURBS curve and its radius of curvature plots
- (b) : shape modified NURBS curve and its radius of curvature plots
- (c) : difference between current radius of curvature and specified radius of curvature
- (d) : radius of curvature of shape modified NURBS curve and specified radius of curvature (same as in (c))

5 GENERATION OF A NURBS CURVE USING THE LINE SEGMENTS

The concept of generation of a NURBS curve based on the regression line segments determined using the detected edges is illustrated in Figure 11.

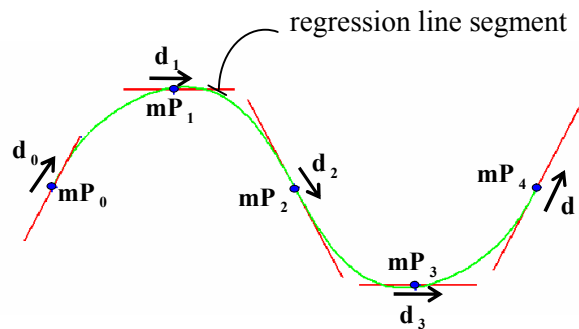


Figure 11. Concept of generation of a NURBS curve which approximates the regression line segments

mP_0, mP_1, \dots, mP_4 are the midpoints of the corresponding regression line segments. d_0, d_1, \dots, d_4 are directional unit vectors of the corresponding line segments. A NURBS curve which passes through the midpoints and has first derivatives which are proportional to the directional unit vectors is generated. Eq.(14) is applied to the midpoints of the line segments by setting the parameter of Eq.(5) to zero.

$$R_i = \frac{1}{120}(q_i + 26q_{i+1} + 66q_{i+2} + 26q_{i+3} + q_{i+4}) \quad (i = 0, 1, \dots, m-1) \quad (14)$$

where m is the number of line segments.

Eq.(15) is applied to the directional unit vectors of the line segments by setting the parameter of Eq.(6) to zero while considering the magnitude of the first derivatives.

$$\frac{dR_i}{dt} = \frac{1}{24}(-q_i - 10q_{i+1} + 10q_{i+3} + q_{i+4}) \quad (i = 0, 1, \dots, m-1) \quad (15)$$

where m is the number of line segments.

A NURBS curve is generated by solving Eq.(14) and Eq.(15) simultaneously. If the number of line segments is 4, the number of NURBS curve equations (Eq.(14)) is 4 and the number of first derivative equations (Eq.(15)) is 4. As a linear system, the total number of equations is 8, whereas the total number of control points of a NURBS curve is 8. Therefore, this linear system is determined. That is, the rank of a coefficient matrix of a linear system is equal to the number of unknowns. The solution to this linear system is exact.

But, in case the number of line segments is 3, the number of equations (Eq.(14)) which pass through the midpoints is 3, and the number of equations of the first derivative (Eq.(15)) is 3. In this case, as a linear system, the number of equations is 6, whereas the number of control points of the NURBS curve is 7. That is, the number of equations is less than the number of unknowns. Therefore, this linear system is underdetermined [31].

For an underdetermined system, while setting auxiliary function, the linear system is solved under the constraint condition by selecting one solution from infinite number of exact solutions using the Lagrange's method of indeterminate multipliers.

In case the number of line segments is 5, the number of equations (Eq.(14)) is 5, and the number of equations of the first derivative (Eq.(15)) is 5. In this case, as a linear system, the number of equations is 10, whereas the number of control points of the NURBS curve is 9. That is, the number of equations exceeds the number of unknowns. Therefore, this linear system is overdetermined [32]. For an overdetermined system, the differences of right and left side of all the equations of the system are minimized. The control points calculated are an approximation.

For a system, where the number of line segments is more than 5, the linear system is overdetermined. For these systems, in accordance with the increment of the difference between the number of equations and the number of unknowns, the status of approximation solution becomes worse.

The above mentioned are summarized in Table 1. A determined linear system is shown by the cross hatching.

Table 1. Linear system condition

(I)	(II)	(III)	(IV)
2	6	Underdetermined	Exact
3	7	Underdetermined	Exact
4	8	Determined	Exact
5	9	Overdetermined	Approximation
6	10	Overdetermined	Approximation
7	11	Overdetermined	Approximation
8	12	Overdetermined	Approximation

- (I) Number of line segments
- (II) Number of control points of a NURBS curve
- (III) System condition (Underdetermined or Overdetermined)
- (IV) Solution status

As an example, in case the number of line segments is 3, that is, m in Eq.(14) and in Eq.(15) is 3, the NURBS curve generated as an underdetermined system is shown in Figure 12 with its first derivative, which is drawn outward perpendicular to the curve by the straight lines. The length of the line is proportional to the first derivative. This is an unusual way of displaying the first derivatives. Nevertheless, this helps visual recognition of the NURBS curve shape and its first derivative magnitude variation. The solution to this linear system is exact.

In case the number of line segments is 4, that is, m in Eq.(14), and in Eq.(15) is 4, the NURBS curve generated as a determined system is shown with its first derivative vectors in Figure 13. The solution to this linear system is exact.

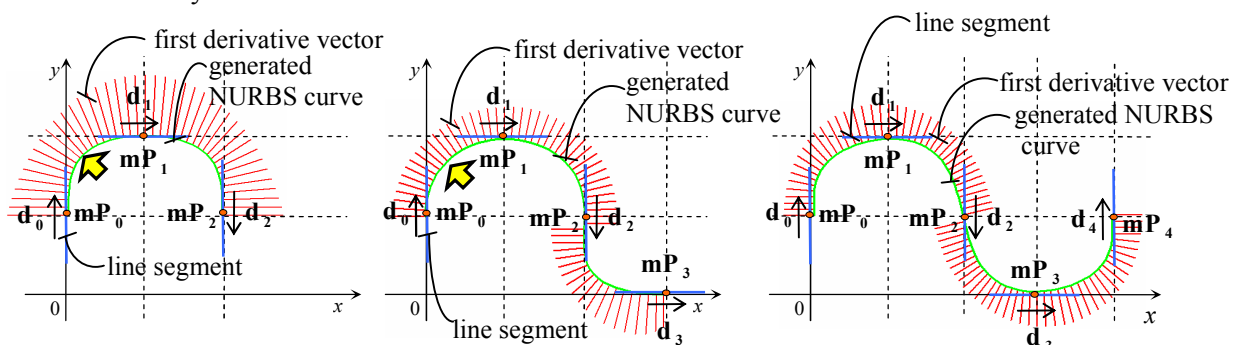


Figure 12. NURBS curve and its first derivative vectors, in case of underdetermined

Figure 13. NURBS curve and its first derivative vectors, in case of determined

Figure 14. NURBS curve and its first derivative vectors, in case of overdetermined

Moreover, in case the number of line segments is 5, that is, m in Eq.(14) and in Eq.(15) is 5, the NURBS curve generated as an overdetermined system is shown with its first derivative vectors in Figure 14. The solution to this linear system is an approximation. In this manner, a NURBS curve is generated based on the line segments.

The shape of the curve indicated by the arrow in Figure 12 and that of the curve indicated by the arrow in Figure 13 are visually different. This is caused by the difference of the magnitude of the first derivative given to Eq.(15). At the same time, the shape around mp_2 in Figure 13 and mp_2 in Figure 14 look visually different. As mentioned above, the shape of the curve is different, the position which should be passed is different, and the direction of the first derivative is different. These differences are acceptable for the purpose of our study which is extracting a NURBS curve using a hand-drawn sketch.

6 EXAMPLES OF NURBS CURVE GENERATION BASED ON A HAND-DRAWN SKETCH

Using the techniques mentioned in the previous sections, a NURBS curve generation process illustrated in Figure 15 is described according to the following steps.

The hand-drawn sketch image shown in Figure 15(a) is put into a computer by using a scanner set at 120 dots per inch. A gradated background is shown in this hand-drawn sketch. Therefore, transformation to a logarithmic image is necessary. The logarithmic image is shown in Figure 15(b).

Then, the hand-drawn sketch is binarized. The binary hand-drawn sketch image is shown in Figure 15(c). The edge of the binary sketch is smoothed by applying feature based erosion and dilation. The objective of this process is to smooth the outer edge of the binary image. Therefore, internal image contours are eliminated.

The edge smoothed binary sketch image is shown in Figure 15(d). Then applying the Laplacian operation, the edges of the smoothed binary sketch image are detected. The detected edges indicated by dots are shown with the sketch image as shown in Figure 15(e). Using the detected edges, regression line segments are determined.

These line segments are shown in Figure 15(f). Using the determined line segments, a quintic NURBS curve is generated.

The generated NURBS curve with radius of curvature plots is shown in Figure 15(g).

Radius of curvature distribution and specified radius of curvature according to the designer's intention are shown in Figure 15(h). The curve shape modification algorithm is then applied to this NURBS curve.

A shape modified quintic NURBS curve is shown with its radius of curvature plots in Figure 15(i).

Radius of curvature distribution of a shape modified NURBS curve is shown with the specified radius of curvature distribution in Figure 15(j).

Thus, following these steps, a NURBS curve according to the designer's intention can be generated.

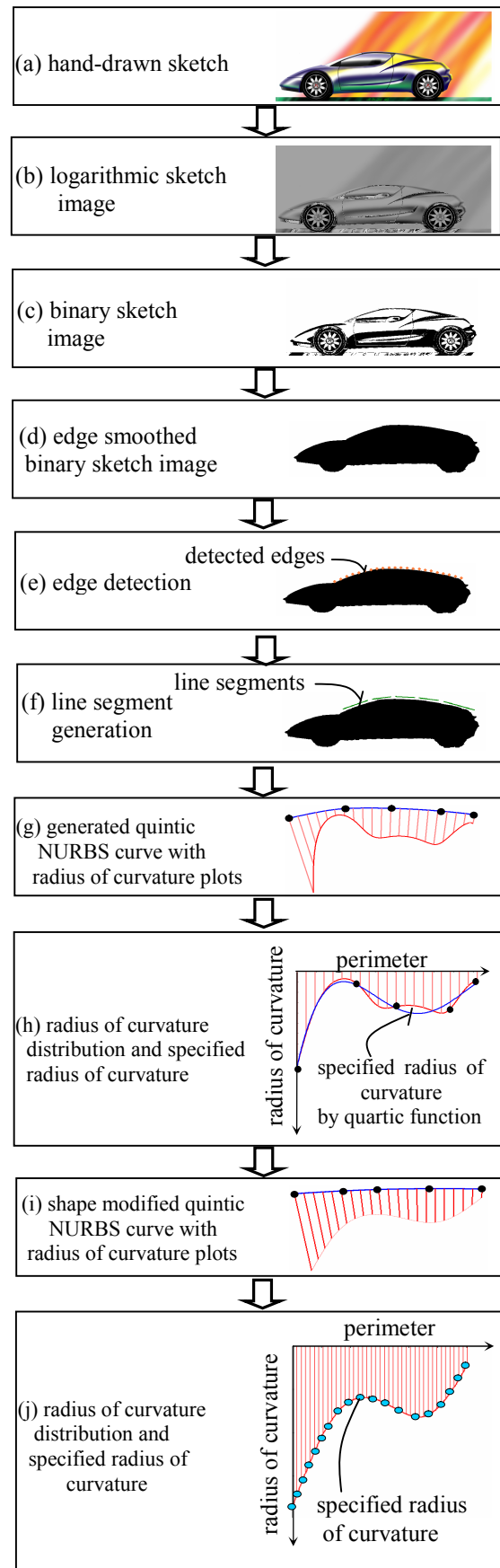


Figure 15. Illustration for the steps from data input to generation of shape modified NURBS curve

7 CONCLUSIONS

The objective of this study is to shorten the product design period by extracting a freeform curve from a hand-drawn sketch. Edge smoothing of the binary sketch image by feature based erosion and dilation, and line segment generation using the detected edges are described. A freeform curve is expressed as a quintic NURBS curve, and NURBS curve generation based on the line segments are described. Then, a method to modify a NURBS curve shape based on the specified radius of curvature distribution is described. Examples of NURBS curve generation based on a hand-drawn sketch are given according to the generation steps.

We have proposed an edge smoothing method of the binary image by feature based erosion and dilation. We have also proposed a method to generate a NURBS curve using the given points and the gradient on the line segments simultaneously.

Moreover, we have also proposed a method to modify the shape of the NURBS curve according to the specified radius of curvature distribution.

A definition of a fair curve and specification of the radius of curvature are issues to be examined in the future.

REFERENCES

- [1] Alias Learning Tools, "Learning Design with Alias StudioTools: A Hands-on Guide to Modeling and Visualization in 3D (Official Alias Training Guide)", Sybex; Pap/Dvdr edition (2006).
- [2] 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.
- [3] S. Saga, "A freehand interface for computer aided drawing systems based on the fuzzy spline curve identifier", *IEEE*, (1995), pp.2754-2759.
- [4] T. Igarashi, S. Matsuoka, H. Tanaka, "Teddy: A sketching interface for 3D freeform design", *SIGGRAPH 99*, (1999), pp.409-416.
- [5] 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.
- [6] 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.
- [7] K. Matsuda, S. Sugishita, K. Kondo, H. Sato, and S. Shimada, "Freehand sketch system for 3D geometric modeling", *IEEE*, (1997), pp.55-62.
- [8] 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*", (2002), pp.579-587.
- [9] T.A. Galyean, J.F. Hughes, "Sculpting: An interactive volumetric modeling technique", *Proc. ACM SIGGRAPH '91*, Vol.25, Num 4, (1991), pp.267-273.
- [10] R.C. Zeleznik, K.P. Herndon, J.F. Hughes, "SKETCH: An interface for sketching 3Dscenes" *Computer graphics Proceedings*, (1996), pp.163-170.
- [11] 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*, (2003), pp.613-624.
- [12] A. Shesh, and B. Chen, "SMARTPAPER: An interactive and user friendly sketching system", *EUROGRAPHICS 2004*, vol.23 (2004), pp.301-310.
- [13] E. Schweikardt, M.D. Gross, "Digital clay: deriving digital models from free hand sketches", *Automation in Construction*, 9, (2000), pp.107-115.
- [14] C. Zhang, P. Zhang, F. (F). Cheng, "Fairing spline curves and surface by minimizing energy", *Computer Aided Design*, 33, (2001), pp.913-923.
- [15] X. Yang, G. Wang, "Planar point set fairing and fitting by arc splines", *Computer Aided Design*, 33, (2001), pp.35-43.
- [16] W.H. Frey, D.A. Field, "Designing Bézier conic segments with monotone curvature", *Computer Aided Geometric Design*, 17, (2000), pp.457-483.

- [17] 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.
- [18] 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.
- [19] H. Nowacki, D. Liu, X. Lu, "Fairing Bézier curves with constraints", *Computer Aided Geometric Design*, 7, (1990), pp.43-55.
- [20] 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.
- [21] L. Fang, D.C. Gossard, "Multidimensional curve fitting to unorganized data points by nonlinear minimization", *Computer Aided Design*, 27, 1, (1995), pp.48-58.
- [22] J. Unger, "Rendering in mixed media", Watson-Guptill Publications, (1985).
- [23] J.C. Russ, *The image processing handbook forth edition*, CRC Press, Florida, (2002), pp.397-416.
- [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. S. Bonn, W. G. Bowman, *McGraw-Hill Dictionary of Scientific and technical terms Sixth Edition*. McGraw-Hill, (2002), pp.2054.
- [27] C. de Boor, "On calculating with B-spline", *Japprox. Theory*, 6(1), (1972), pp.50-62.
- [28] D. Marsh, "Applied Geometry for Computer Graphics and CAD", Springer-Verlag, (2005), pp.188.
- [29] Eugene V. Shikin.: *Hand Book and Atlas of CURVES*. CRC Press BocaRaton, Florida, (1995), pp.29.
- [30] L. Piegl, W. Tiller, "The NURBS Book", Springer-Verlag, (1997), pp.510.
- [31] W. Boehm, and H. Prautzsch, "Geometric Concepts for Geometric Design", (1994), pp.26-27.
- [32] G. Farin, and D. Hansford, "Practical Linear Algebra –A Geometry Toolbox –", (2005), pp.253-254.

Contact: Akira Yamaguchi
 Meisei University
 Graduate School of Information Science
 2-590 Nagabuchi, Ome-City, Tokyo
 Japan
 Tel: Int + 81 428 25 5225
 Fax: Int + 81 428 25 5184
 Email: 05dj002@edu.meisei-u.ac.jp