Surface Fairing with Boundary Continuity Based on the Wavelet Transform

Joohyung Cho, Tae-wan Kim, and Kunwoo Lee

The surface modeling capability of CAD systems is widely used to design products bounded by free form surfaces and curves. However, the surfaces or curves generated by popular data fitting methods usually have shape imperfections such as wiggles. Thus, fairing operations are required to remove the wiggles, which makes the surfaces or curves smooth. This paper proposes a new method based on the wavelet transform for fairing the surfaces or curves while preserving the continuity with adjacent surfaces or curves. The wavelet transform gives a hierarchical perspective of the surfaces and the curves, which can be decomposed into the overall sweep and details, i.e., local deviations from sweep like the wiggles. The proposed fairing method provides a similar effect on the mathematical surface as that of the grinding operation using sandpaper on the physical surface.

Manuscript received June 27, 2000; revised May 7, 2001.
This work was supported by the Korean Ministry of Science and Technology through the National Research Laboratory for Modeling Kernel Development.
Joohyung Cho is with the Daewoo Heavy Industries & Machinery, Incheon, Korea.
(phone: +82 32 760 6155, e-mail: max@solar.dhiltd.co.kr)
Tae-wan Kim is with the Sejong University, Seoul, Korea.
(phone: +82 2 3408 3795, e-mail: tkim@ sejong.ac.kr)
Kunwoo Lee is with the Seoul National University, Seoul, Korea.
(phone: +82 2 880 7141, e-mail: kunwoo@snu.ac.kr)

I. INTRODUCTION

Currently, hybrid solid/surface modelers are widely used to design products bounded by free form curves and surfaces. These modelers usually provide the function of generating surfaces or curves in many ways such as skinning, sweeping, swinging, lofting, data fitting, interpolation, etc. Although the surfaces or curves generated by these methods satisfy the user's purpose in most cases, there may be some shape imperfections in the surfaces or curves. In computational geometry terminology, shape imperfection means the region which has frequent saddle points in the curve or the frequent variation in the sign of the Gaussian curvature in the surface. Usually, the surfaces or the curves generated by interpolation or data fitting may have local wiggles with high amplitude. In particular, the surface generated by the interpolation and data fitting method from the products on the clay model of an automobile may have severe local wiggles and frequent discontinuity in the curvature. These shape imperfections cause significant economic loss in time and cost in later design stages. Thus, the process called 'fairing' is required to make the surface smooth by removing local wiggles. This paper proposes a new fairing method based on the wavelet transform that preserves continuity across the boundary with neighboring surfaces.

II. RELATED WORKS

The wavelet transform was introduced to replace the windowed Fourier transform in 1980's. Although it has its roots in the field of the approximation theory and signal processing, it is widening its application to areas such as turbulence, radar signals, seismic geology, image compression and computer graphics. In general, the wavelet transform decomposes a function into a coarse overall shape and detail coefficients hierarchically. Thus, the wavelet transform presents a tool, by which an image, a sound signal, surfaces or curves can be analyzed in different levels of detail. It also allows the multiresolution analysis of a function. Some publications in which the wavelet transform is applied to curves and surfaces are summarized as below.

Forsey and Bartels [1] employed hierarchical B-splines to address the problem of editing the overall sweep of a surface while maintaining its details. This construction is similar in spirit to the filter bank process used in multiresolution analysis. One significant difference is that in their formulation there are an infinite number of possible representations for the same surface, whereas the multiresolution curve representation is unique for a given shape. Chui [2], [3] suggested the B-spline wavelet. He improved the traditional Mallat's [4] wavelet, which has global supports, by adapting minimal supports so that local modification is possible. Gene Chuang [5] developed the wavelet descriptor. Using the wavelet transform, the hierarchical planar curve descriptor, i.e., the wavelet descriptor, decomposes a curve into components of different scales so that the coarsest scale components carry the global approximation information while the finer scale components contain the local detailed information. Chuang used the wavelet descriptor in curve matching and shape recognition. Gortler [6] developed variational geometric modeling with wavelets. To realize a more intuitive modeling, he directly manipulated curves and surfaces with a set of constraints such as interpolation and tangent constraints. By using a wavelet basis, the "best" solution which requires solving a variational problem can be obtained more efficiently because the iterative methods converge rapidly owing to the hierarchical nature of the wavelet basis.

For the fairing of B-spline curves and surfaces, Farin and Sapidis [7] used knot removal and insertion technique. Greiner [8] described variational design for fairing. Rando and Roulier [9] suggested designing of parametric surfaces. Moreton and Sequin [10] utilized a functional optimization problem.

Finkelstein [11] developed a multiresolution curve using the B-spline wavelet transform. Using multiresolution analysis of a curve, curve smoothing, curve compression and scan conversion are accomplished. His curve smoothing method will be well described in section III.2 and the surface and curve fairing method proposed in this paper is originally derived from his work. Finkelstein's fairing method works well but there is no consideration on the continuity with neighboring curves or surfaces. Thus, an improved fairing method for surfaces and curves is proposed to guarantee the continuity across the boundary with neighboring surfaces or curves.

III. CURVE FAIRING

1. B-Spline Scaling Functions and Wavelet Functions

Finkelstein [11] suggested the B-spline wavelet transform for the multiresolution analysis of the B-spline curves. For arbitrary type surfaces, Lounsbery *et al.* [12] proposed multiresolution analysis algorithms. In this section, the definition of B-spline scaling functions [13] and wavelet functions are described. We use end point interpolating B-spline scaling functions.

A. B-Spline Scaling Function

The definition of the end point interpolating B-spline functions [14]-[16] of level j is described in (1) and (2):

$$N_{i,0}^{j}(u) = \begin{cases} 1 & \text{if } u_i \le u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}, \tag{1}$$

$$N_{i,p}^{j}(u) = \frac{u-u_i}{u_{i+p}-u_i} N_{i,p-1}^{j}(u) + \frac{u_{i+p+1}-u}{u_{i+p+1}-u_{i+1}} N_{i+1,p-1}^{j}(u), \tag{2}$$

$$[u_0,\ldots,u_{2^j+2p}] = \frac{1}{2^j}[\underbrace{0,\ldots,0}_{p+1\text{ times}},1,2,\ldots,2^j-2,2^j-1,\underbrace{2^j,\ldots,2^j}_{p+1\text{ times}}]. \tag{3}$$

The functions $N_{o,p}^{j}(u),\ldots,N_{n,p}^{j}(u)$ form bases for the space of piecewise-polynomials of degree p of level j. The level represents how many times the vector space, knot vector in this case, can be subdivided. Also, the number of basis functions is related to the number of knots and degree of B-spline scaling function. Thus, the level is dependent upon the number of B-spline scaling functions, that is, the number of control points. Some B-spline scaling functions for V^1 for degrees 0 to 3 are shown in Fig. 1.

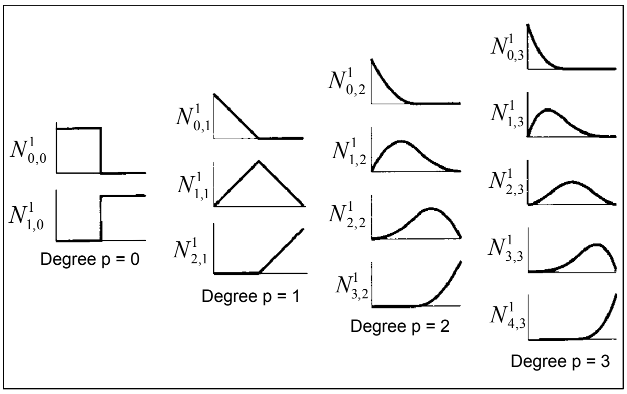

Fig. 1. B-spline scaling functions, level $j = 1$ (courtesy of "wavelets for computer graphics").

B. B-Spline Wavelet Function

The wavelet function is determined by the matrix Q^j, but there is no unique solution for the Q^j matrix. Thus, Finkelstein [11] derived the B-spline wavelet function with additional constraint, that is, wavelet function with small support. Figure 2 shows the cubic B-spline wavelet functions and scaling functions for level $j = 3$.

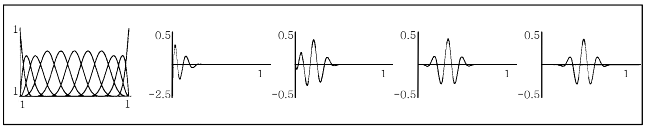

Fig. 2. Cubic B-spline scaling function and wavelet functions ($j = 3$) (courtesy of "wavelets for computer graphics").

2. Curve Smoothing Using the B-Spline Wavelet

The end point interpolating uniform B-spline curve of level j of this curve is described in (4)

$$C^j(u) = \sum_{i=0}^{2^j+p-1} N_{i,p}^j(u) V_i^j \tag{4}$$

p : *degree,*

V_i^j : *control points,*

$N_{i,p}^j$: normalized B-spline basis functions of level j defined on $[u_0, u_1, \ldots, u_{2^j+2p}]$,

$[u_0, u_1, \ldots, u_{2^j+2p}]$: knot vector.

In this case, the number of control points of the curve is $2^j + p$.

The comparison of the general wavelet transform with the B-spline wavelet transform is shown in Table 1.

The scaling function of the general wavelet transform corresponds to the B-spline basis function, and the coefficients of the scaling function in the general wavelet transform correspond to the control points in the B-spline wavelet transform. Fairing a j level curve into a lower resolution curve, i.e., $j-1$ level curves using the wavelet transform, is accomplished by finding new control points V_i^{j-1} of the $j-1$ level curve from the control points V_i^j of the j level curve using the filter bank.

The new $j-1$ level curve is described in (5).

$$C^{j-1}(u) = \sum_{i=0}^{2^{j-1}+p-1} N_{i,p}^{j-1}(u) V_i^{j-1} \tag{5}$$

Lowering the level of the curve results in a decrease in the number of control points. Missing the detail coefficients, the curve shows its overall shape rather than its local features. Figure 3 shows the process of fairing the curve.

The curve faired by this algorithm is shown in Figs. 4 to 7. The level of the original curve is 5. In these figures, the dashed line represents the control polygon of the curve and each dot represents a control point of the curve. Severe wiggles are shown in the original curve, but as the level is lowered, the amplitude of the wiggle is reduced and the overall shape is smoothly faired.

Also, we can see the number of control points is decreased as the level is lowered. So, we can deduce from this example that the detail coefficients of the curve imply local wiggles of the curve. Consequently, the original curve can be decomposed into lower level curves representing the overall shape of the curve and detail coefficients, that is, wiggles of the curve by the wavelet transform.

Table 1. General wavelet vs. B-spline wavelet.

General Wavelet	B-spline Wavelet
$f^j(u) = \Phi^j(u) C^j$	$C^j(u) = \sum_{i=0}^{2^j+p-1} N_{i,p}^j(u) V_i^j$
$\Phi^n(u)$: scaling function	$N_{i,p}^j(u)$: B-spline basis function
C^j : coefficient	V_i^j : control point

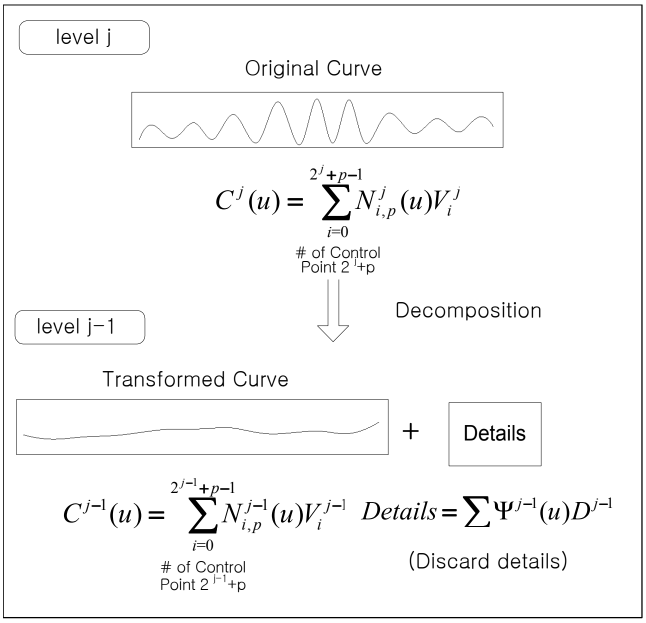

Fig. 3. Curve fairing using the wavelet transform.

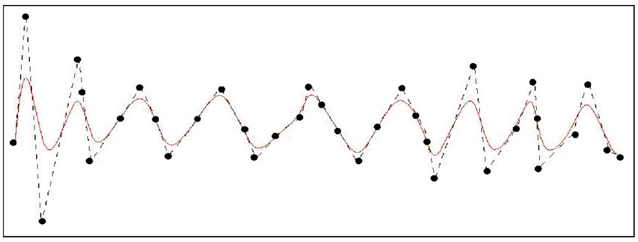

Fig. 4. Original curve ($j = 5$).

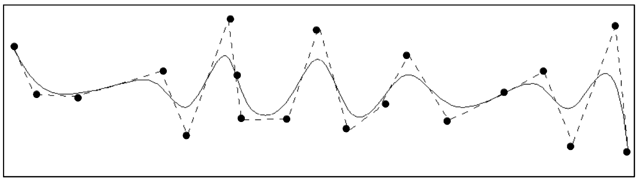

Fig. 5. Faired curve $(j = 4)$.

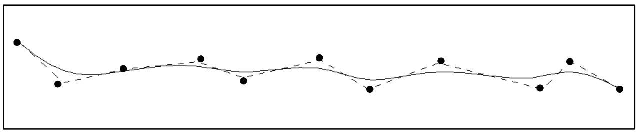

Fig. 6. Faired curve $(j = 3)$.

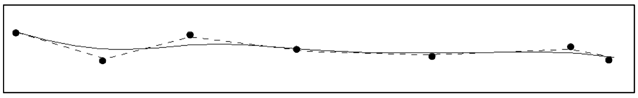

Fig. 7. Faired curve $(j = 2)$.

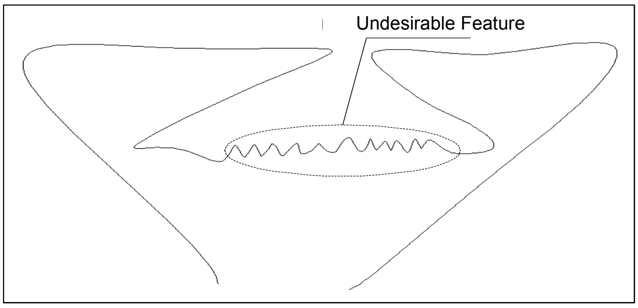

Fig. 8. Original curve (wiggles in the middle of the curve).

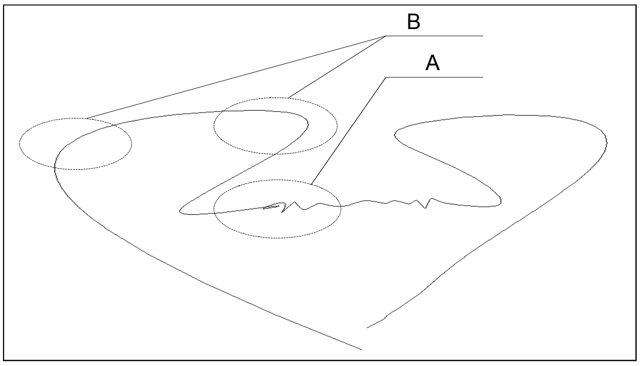

Fig. 9. Faired curve (one level lowered).

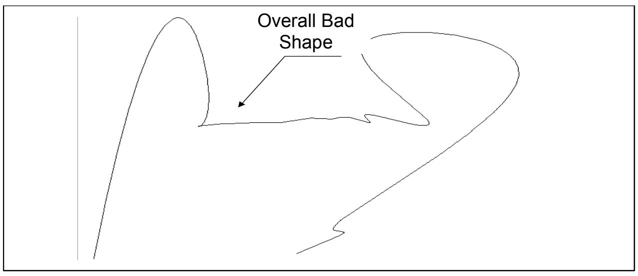

Fig. 10. Faired curve (two level lowered).

3. Problems in the B-Spline Wavelet Transform

Figure 8 illustrates an example where local wiggles are concentrated in the middle of the curve. To remove these local wiggles, the curve is faired and the results are shown in Figs. 9 and 10.

Consider the curve lowered by one level by the wavelet transform. Although the amplitudes of the local wiggles are reduced, there is unexpected self-intersection in the region indicated by A where the region of local wiggles is connected to the other region. Besides, there are unexpected faired regions indicated by B which should not have been faired. The curve in Fig. 10 which is wavelet transformed once again has an undesirable overall shape. This curve is transformed to an undesirable shape because the magnitudes of the detail coefficients of the wiggle region are different from those of the non-wiggle region. In the mixed region where the region of local wiggles are connected to the other region, the detail coefficients are mixed with each other: i.e., not only the detail coefficients of the wiggle region but also those of the non-wiggle region are transformed together.

As described above, the curve can be faired by the wavelet transform which removes undesirable wiggles in the curve. However, the curve is faired desirably only in the case where the wiggles of the curve are relatively uniform. If the wiggles of the curve are concentrated locally, the transformed curve might have an overall bad shape. To avoid this problem, it must be allowed to fair the local region selected by a user. Also, after fairing, the selected region should be connected smoothly to the remaining non-selected region. However, the curve transformed by the algorithm in section III.2 cannot preserve the continuity at the end points. Figure 11 illustrates the result after fairing the curve in Fig. 4. It shows that the end points of the faired curve are not coincident with those of the original curve. To solve this problem, a local wavelet transform to fair a portion of the curve is suggested in this paper.

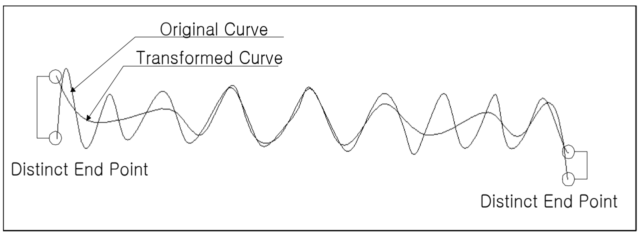

Fig. 11. Original curve and faired curve.

4. Wavelet Transform Preserving Continuity at the Ends

As mentioned in the previous section, the wavelet transform of a B-spline curve can not preserve the continuity at the end points. The reason is as follows.

Consider the wavelet transform of a j level curve. The j level curve is decomposed into $j-1$ level curves and detail coefficients which are the coefficients of wavelet functions of $j-1$ level as shown in (6) and (7).

$$\begin{aligned} C^j(u) &= \sum_{i=0}^{2^j+p-1} N_{i,p}^j(u)V_i^j \\ &= \sum_{i=0}^{2^{j-1}+p-1} N_{i,p}^{j-1}(u)V_i^{j-1} + \sum_{i=0}^{2^{j-1}} \Psi_{i,p}^{j-1}(u)D_i^j \\ &= C^{j-1}(u) + \text{Details}. \end{aligned} \tag{6}$$

$$C^{j-1}(u) = \sum_{i=0}^{2^{j-1}+p-1} N_{i,p}^{j-1}(u)V_i^{j-1} . \tag{7}$$

When the knot vector is normalized, the function values at $u=0$ and $u=1$ of the j level curve should be the same as those of the $j-1$ level curve to preserve C^0 continuity at both end points during the fairing process.

As the B-spline curve has compact support, the number of nonzero B-spline basis functions is only $p+1$ for any knot span $[u_{i-1}\ u_i]$. So, for the j level curve, the maximum number of nonzero B-spline basis functions at $u=0$, i.e., for the knot span $[u_0\ u_1]$, is p. The function value of the j level curve at $u=0$ is described as follows:

$$\begin{aligned} C^j(0) &= N_{0,p}^j(0)V_0^j + N_{1,p}^j(0)V_1^j \\ &\quad + N_{2,p}^j(0)V_2^j + \ldots + N_{p-1,p}^j(0)V_{p-1}^j \end{aligned} \tag{8}$$

The detail coefficients derived from the j level curve are expressed in terms of linear combinations of $j-1$ level B-spline wavelet basis functions. These detail coefficients can be expressed in terms of linear combinations not of $j-1$ level B-spline wavelet basis functions but of one level higher, j level B-spline basis functions, using the Q^j matrix for the synthesis filter bank.

$$\Psi_{i,p}^{j-1}(u) = N_{i,p}^j(u)Q^j . \tag{9}$$

When the detail coefficients are expressed in the form of j level B-spline basis functions, the nonzero component of the B-spline basis function at $u=0$ is shown as follows. In (10), $\widetilde{D}_i^j$ represents the i-th resultant details coefficient, which is multiplied by the Q^j matrix.

$$\begin{aligned} \sum \Psi_{i,p}^{j-1}(0)D_i^{j-1} &= \sum N_{i,p}^j(0)Q^jD_i^{j-1} \\ &= N_{0,p}^j(0)\widetilde{D}_0^{j-1} + N_{1,p}^j(0)\widetilde{D}_1^{j-1} \\ &\quad + N_{2,p}^j(0)\widetilde{D}_2^{j-1} + \ldots + N_{p-1,p}^j(0)\widetilde{D}_{p-1}^{j-1}. \end{aligned} \tag{10}$$

If (10) is applied to (6), the function value of the j level curve at $u=0$ is derived in (11).

$$\begin{aligned} C^j(0) &= N_{0,p}^j(0)V_0^j + N_{1,p}^j(0)V_1^j \\ &\quad + N_{2,p}^j(0)V_2^j + \ldots + N_{p-1,p}^j(0)V_{p-1}^j \\ &= C^{j-1}(0) + \sum \Psi_{i,p}^{j-1}(0)D_i^{j-1} \\ &= C^{j-1}(0) + \sum N_{i,p}^j(0)Q^jD_i^{j-1} \\ &= N_{0,p}^{j-1}(0)V_0^{j-1} + N_{1,p}^{j-1}(0)V_1^{j-1} \\ &\quad + N_{2,p}^{j-1}(0)V_2^{j-1} + \ldots + N_{p-1,p}^{j-1}(0)V_{p-1}^{j-1} \text{ - (a)} \\ &\quad + N_{0,p}^j(0)\widetilde{D}_0^{j-1} + N_{1,p}^j(0)\widetilde{D}_1^{j-1} \\ &\quad + N_{2,p}^j(0)\widetilde{D}_2^{j-1} + \ldots + N_{p-1,p}^j(0)\widetilde{D}_{p-1}^{j-1} \text{ - (b)} \end{aligned} \tag{11}$$

Additionally, the function value of the $j-1$ level curve at $u=0$ is shown in (12).

$$\begin{aligned} C^{j-1}(0) &= N_{0,p}^{j-1}(0)V_0^{j-1} + N_{1,p}^{j-1}(0)V_1^{j-1} \\ &\quad + N_{2,p}^{j-1}(0)V_2^{j-1} + \ldots + N_{p-1,p}^{j-1}(0)V_{p-1}^{j-1} \end{aligned} \tag{12}$$

When the j level curve is wavelet-transformed, all detail coefficients are removed, i.e., part (b) in (11). As a result, C^0 continuity of the curve at $u=0$ cannot be preserved.

$$C^j(u=0) \neq C^{j-1}(u=0) . \tag{13}$$

In the same manner, C^0 continuity of the curve at $u=1$ cannot be preserved either.

$$C^j(u=1) \neq C^{j-1}(u=1) . \tag{14}$$

For the reason mentioned above, part (b) in (11) which influences the function value of the end points should not be removed and should be merged into the $j-1$ level curve to preserve continuity during the wavelet transform.

However, in this case, the detail coefficients are expressed in the form of j level B-spline functions, and therefore, these cannot be merged into a $j-1$ level curve. Thus, new knots should be inserted into the knot vectors of the $j-1$ level curve using the Boehm algorithm so that the knot vector and the number of control points of this curve should be the same as those of the one level higher j level curve. This knot insertion operation of a cubic B-spline of level j is shown in Fig. 12.

$$\boxed{\frac{1}{2^j}}\ \boxed{\frac{3}{2^j}}\ \boxed{\frac{5}{2^j}}\ \cdots\ \boxed{\frac{2^j-1}{2^j}}$$

$$U^{j-1} = \left\{0,0,0,0,\frac{1}{2^{j-1}},\frac{2}{2^{j-1}},\frac{3}{2^{j-1}},\ldots,\frac{2^{j-1}-1}{2^{j-1}},1,1,1,1\right\}$$

$$\Longrightarrow U^j = \left\{0,0,0,0,\frac{1}{2^j},\frac{2}{2^j},\frac{3}{2^j},\ldots,\frac{2^j-1}{2^j},1,1,1,1\right\}$$

Fig. 12. Raising the level by knot insertion.

Then, the detail coefficients are added into the corresponding control points to generate a new j level curve preserving C^0 continuity at the end points. Figure 12 shows the knot insertion process.

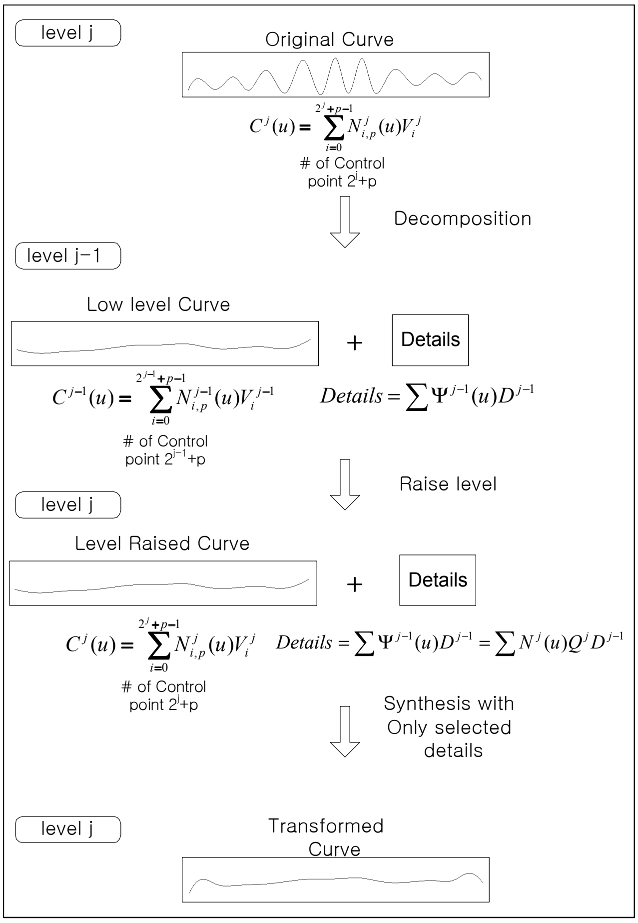

Fig. 13. Fairing algorithm preserving C^0 continuity by the wavelet transform.

Figure 13 shows a fairing algorithm in which the C^0 continuity is preserved by the wavelet transform. The original curve is decomposed with low level curve and details. The level is then raised and only selected details are added into the wavelet transformed curve in the synthesis process.

The same fairing approach can be applied to preserve not only C^0 continuity but also C^n continuity. However, in this case, the part to be merged into the $j-1$ level curve is the detail coefficients which influence n-th derivatives of the j level curve at the end points. So, merging these detail coefficients can generate a new faired curve preserving C^n continuity at the end points. A user can assign the order of continuity at the end points while fairing by this algorithm. A new faired curve transformed by this process is shown in Fig. 14. In Fig. 14, the faired curve preserves C^1 continuity at the end points, that is, it is tangent to the original curve at the end points.

The comparison of the faired curve by the traditional wavelet transform with the faired curve by the proposed wavelet transform is shown in Fig. 15. Except the region near the end points, the faired curve by the proposed wavelet transform is identical with the one by the original wavelet transform.

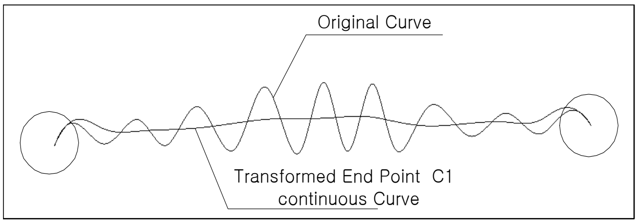

Fig. 14. Fairing curve preserving continuity.

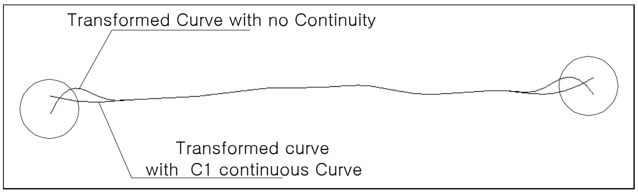

Fig. 15. Comparison of the traditional wavelet transform and the suggested wavelet transform.

IV. SURFACE FAIRING

Basically, a B-spline surface is a tensor product of the B-spline curve and there are two directions u, v in the parametric domain. The definition of the B-spline surface is given in (15). Like the case of the B-spline curve, we use end point interpolating B-spline surfaces. The u level of the surface is n, and the v level is m. The degrees of the surface in the u and v directions are p and q, respectively.

$$S(u,v) = \sum_{i=0}^{2^n+q-1} \sum_{j=0}^{2^m+p-1} N_{i,p}^n(u)N_{j,q}^m(v)V_{i,j}\ . \tag{15}$$

1. The Derivatives of the B-Spline Surface

The general definition of the derivatives of a B-spline surface is shown in (16) and (17).

$$\frac{\partial^{k+l}}{\partial^k u \partial^l v} S(u,v) = \sum_{i=0}^{2^n+q-k-1} \sum_{j=0}^{2^m+p-l-1} N_{i,p-k}^n(u)N_{j,q-l}^m(v)V_{i,j}^{(k,l)} \tag{16}$$

$$V_{i,j}^{(k,l)} = (q-l+1)\frac{V_{i,j}^{(k,l-1)} - V_{i,j-1}^{(k,l-1)}}{v_{j+q-1} - v_{j-l}}\ . \tag{17}$$

Therefore, the first derivative and twist vector of the B-spline surface are derived as shown in (18) through (23).

$$S_u(u,v) = \sum_{i=0}^{2^n+q-2} \sum_{j=0}^{2^m+p-1} N_{i,p-1}^n(u) N_{j,q}^m(v) V_{i,j}^{(1,0)} \quad (18)$$

$$V_{i,j}^{(1,0)} = p \frac{V_{i+1,j} - V_{i,j}}{u_{j+p+1} - u_{i+1}} \quad (19)$$

$$S_v(u,v) = \sum_{i=0}^{2^n+q-1} \sum_{j=0}^{2^m+p-2} N_{i,p}^n(u) N_{j,q-1}^m(v) V_{i,j}^{(0,1)} \quad (20)$$

$$V_{i,j}^{(0,1)} = q \frac{V_{i,j+1} - V_{i,j}}{v_{j+q+1} - v_{j+1}} \quad (21)$$

$$S_{uv}(u,v) = \sum_{i=0}^{2^n+q-2} \sum_{j=0}^{2^m+p-2} N_{i,p-1}^n(u) N_{j,q-1}^m(v) V_{i,j}^{(1,1)} \quad (22)$$

$$V_{i,j}^{(1,1)} = q \frac{V_{i,j+1}^{(1,0)} - V_{i,j}^{(1,0)}}{v_{j+q+1} - v_{j+1}}. \quad (23)$$

2. Fairing of the B-Spline Surface

Since B-spline surface is a tensor product of the B-spline curve, the algorithm used in fairing the B-spline curve can also be used in fairing the B-spline surface. However, because there are two directions u, v in a B-spline surface, a two dimensional wavelet transform should be applied in fairing a B-spline surface. The two dimensional wavelet transform is the extension of the one dimensional wavelet transform and there are two types of two dimensional wavelet transform as follows.

A. Standard Wavelet Transform

The standard wavelet transform begins with applying the one dimensional wavelet transform to the surface in the u direction first. That is, as shown in Fig. 16, transforms each set of control points of the surface in the u direction respectively.

This will yield a new surface with one level lower in the u direction which has lost details during this operation. Next, we apply the one dimensional wavelet transform for a new surface in the u direction again until the desired u level of the surface. Then apply one dimensional wavelet transform to resultant surface in v direction. Like the wavelet transform in the u direction, wavelet transform in the v direction gives a new surface with one level lower in the v direction which has lost details. Until the v level of the surface reaches the desired level, the surface is transformed in the v direction. Finally it gives a single faired surface, that is, the overall sweep surface and detail coefficients lost in this procedure as shown in Fig. 17. The rectangle represents the simplified B-spline surface and the vertex represents the control points of the B-spline surface.

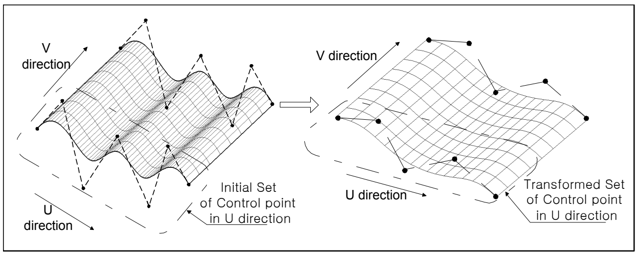

Fig. 16. Fairing the surface in u direction.

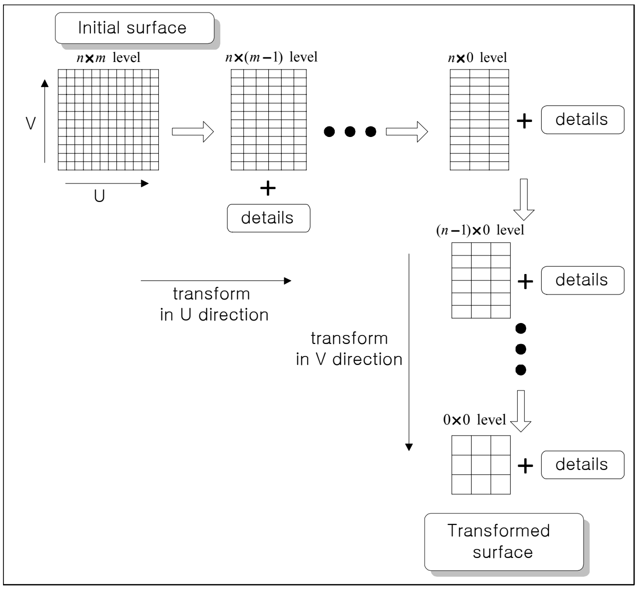

Fig. 17. Standard wavelet transform of simplified B-spline surface.

B. Nonstandard Wavelet Transform

In contrast to the standard wavelet transform, the nonstandard wavelet transform alternates between u and v directions. First, one step of the u directional transform is performed and it gives a new surface with one level lower in the u direction. Next, one step of the v directional transform is applied to the new surface. To complete the transform, we repeat this process in an alternating manner.

The nonstandard wavelet transform is slightly more efficient than the standard wavelet transform in the aspect of computation time. However, it requires that the u level of the surface is the same as the v level of the surface, that is, the same number of control points is required in each direction. Thus, it cannot be applied to the general surface. In this paper, we adapted the standard wavelet transform for surface fairing.

Equation (24) shows the first step of the standard wavelet transform. Assume that the u level of the original surface is n and the v level is m.

$$S^{n,m}(u,v)=\sum_{i=0}^{2^n+p-1}\sum_{j=0}^{2^m+q-1}N_{i,p}^{n}(u)N_{j,q}^{m}(v)V_{i,j}^{n,m} \tag{24}$$

$$=\sum_{j=0}^{2^m+q-1}N_{j,q}^{m}(v)\left\{\sum_{i=0}^{2^n+p-1}N_{i,p}^{n}(u)V_{i,j}^{n,m}\right\} \tag{25}$$

$$=\sum_{j=0}^{2^m+q-1}N_{j,q}^{m}(v)\left\{\sum_{i=0}^{2^{n-1}+p-1}N_{i,p}^{n-1}(u)V_{i,j}^{n-1,m}+\sum_{i=0}^{2^{n-1}}\Psi_{i,p}^{n-1}(u)D_{i,j}^{n-1,m}\right\} \tag{26}$$

$$=\sum_{j=0}^{2^m+q-1}N_{j,q}^{m}(v)\left\{\sum_{i=0}^{2^{n-1}+p-1}N_{i,p}^{n-1}(u)V_{i,j}^{n-1,m}+\sum_{i=0}^{2^n+p-1}N_{i,p}^{n}(u)Q^nD_{i,j}^{n-1,m}\right\} \tag{27}$$

$$=\sum_{i=0}^{2^{n-1}+p-1}\sum_{j=0}^{2^m+q-1}N_{i,p}^{n-1}(u)N_{j,q}^{m}(v)V_{i,j}^{n-1,m}+\sum_{i=0}^{2^n+p-1}\sum_{j=0}^{2^m+q-1}N_{i,p}^{n}(u)N_{j,q}^{m}(u)Q^nD_{i,j}^{n-1,m}. \tag{28}$$

First, let's transform the surface in the u direction. As shown in (24), the surface has u,v directions. The term in parentheses in (25) corresponds to the B-spline curve in the u direction. So, one dimensional wavelet transform in the u direction is accomplished as described in section III.2 using the filter bank algorithm. Then, the u level of the surface is lowered and the details are stored in the latter term in (26). However, because the details can be expressed in terms of the B-spline wavelet function, it can be expressed in terms of a one level higher B-spline scaling function using the synthesis matrix Q^j as shown in (27) and (28), i.e., the original surface is decomposed into one level lower surface which describes the overall sweep of the surface in the u direction and details. We repeat this process to the desired u level, and then transform the surface in the v direction. Finally, the surface can be expressed as the sum of one overall sweep surface and all the other detail coefficients as shown in (29). Then, to fair the original surface, the detail coefficients which represent the deviation from the sweep, that is, the wiggle, should be removed. However, similar to the curve case, the coordinates of the boundary of the surface are changed if the detail coefficients which influence the boundary are simply removed. So, the continuity with the neighboring surfaces across the boundary curves cannot be preserved. In the next section, we present a method to manipulate detail coefficients to preserve continuity.

$$\begin{aligned}S^{n,m}(u,v)&=\sum_{i=0}^{2^n+p-1}\sum_{j=0}^{2^m+q-1}N_{i,p}^{n}(u)N_{j,q}^{m}(v)V_{i,j}^{n,m}\\
&=\sum_{i=0}^{2^0+p-1}\sum_{j=0}^{2^0+q-1}N_{i,p}^{0}(u)N_{j,q}^{0}(v)V_{i,j}^{0,0}\\
&\quad+\sum_{i=0}^{2^0+p-1}\sum_{j=0}^{2^1+q-1}N_{i,p}^{0}(u)N_{j,q}^{1}(v)Q^1D_{i,j}^{0,0}\\
&\quad+\sum_{i=0}^{2^0+p-1}\sum_{j=0}^{2^{2-1}+q-1}N_{i,p}^{0}(u)N_{j,p}^{2}(v)Q^2D_{i,j}^{0,1}\\
&\quad+\ldots+\sum_{i=0}^{2^0+p-1}\sum_{j=0}^{2^{m-1}+q-1}N_{i,p}^{0}(u)N_{j,q}^{m}(v)Q^mD_{i,j}^{0,m-1}\\
&\quad+\sum_{i=0}^{2^0+p-1}\sum_{j=0}^{2^m+q-1}N_{i,p}^{1}(u)N_{j,q}^{m}(v)Q^1D_{i,j}^{0,m}\\
&\quad+\sum_{i=0}^{2^1+p-1}\sum_{j=0}^{2^m+q-1}N_{i,p}^{2}(u)N_{j,p}^{m}(v)Q^2D_{i,j}^{1,m}+\ldots\\
&\quad+\sum_{i=0}^{2^n+p-1}\sum_{j=0}^{2^m+q-1}N_{i,p}^{n}(u)N_{j,p}^{m}(v)Q^nD_{i,j}^{n-1,m}.\end{aligned} \tag{29}$$

3. Manipulating Detail Coefficients

To preserve the continuity when fairing a B-spline surface, some manipulation of the detail coefficients is needed. First, consider the continuity across the boundary. When the knot vector is normalized, equation $u=0$ represents one of the boundaries along the v direction. The first derivative across curve $u=0$ and the twist vector at $u=0$ are shown in (30) through (35). According to these equations, the nonzero component of the B-spline basis function at $u=0$, i.e., for the span of knot vector $[u_0\,u_q]$ are only $N_{0,q-2}^{m}(0),\ldots,N_{q-2,q-2}^{m}(0)$ and the corresponding control points are $V_{i,0},\ldots,V_{i,q-2}$ and $V_{i+1,0},\ldots,V_{i+2,q-2}$. These control points are $q-1$ rows of control points near the boundary along the u direction as indicated in Fig. 18. As $q-1$ rows of control points near the boundary along the u direction influence the first derivatives across the boundary, these $q-1$ directional curves should not be transformed. In the case of the cubic B-spline surface, two $(q=3)$ rows of control points near the boundary should not be transformed. Similarly, to preserve continuity and the twist vector across the curve corresponding to $u=1$, the $q-1$ rows of control points from the boundary should not be transformed. Similarly at $v=0$ and $v=1$, the $p-1$ sets of control points from the boundary also should not be transformed.

$$S_v(u,0)=\sum_{i=0}^{n}\sum_{j=0}^{m-1}N_{i,p}^{n}(u)N_{j,q-1}^{m}(0)V_{i,j}^{(0,1)}$$

$$= \sum_{i=0}^{n} N_{i,p}^{n}(u) \left[\sum_{j=0}^{m-1} N_{j,q-1}^{m}(0) V_{i,j}^{(0,1)} \right]$$

$$= \sum_{i=0}^{n} N_{i,p}^{n}(u) \left[N_{0,q-1}^{m}(0) V_{i,0}^{(0,1)} + N_{1,q-1}^{m}(0) V_{i,1}^{(0,1)} + \ldots + N_{q-2,q-1}^{m}(0) V_{i,q-2}^{(0,1)} \right] \tag{30}$$

$$V_{i,0}^{(0,1)} = q \frac{V_{i,1} - V_{i,0}}{v_{q+1} - v_1} \tag{31}$$

$$V_{i,q-2}^{(0,1)} = q \frac{V_{i,q-1} - V_{i,q-2}}{v_{2q-1} - v_{q-1}} \tag{32}$$

$$S_{uv}(u,0) = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} N_{i,p-1}^{n}(u) N_{j,q-1}(0) V_{i,j}^{(1,1)}$$

$$= \sum_{i=0}^{n-1} N_{i,p-1}^{n}(u) \left[\sum_{j=0}^{m-1} N_{j,q-1}^{m}(0) V_{i,j}^{(1,1)} \right]$$

$$= \sum_{i=0}^{n-1} N_{i,p-1}^{n}(u) \left[N_{0,q-1}^{m}(0) V_{i,0}^{(1,1)} + N_{1,q-1}^{m}(0) V_{i,1}^{(1,1)} + \ldots + N_{q-2,q-1}^{m}(0) V_{i,q-2}^{(1,1)} \right] \tag{33}$$

$$V_{i,0}^{(1,1)} = q \frac{V_{i,1}^{(1,0)} - V_{i,0}^{(1,0)}}{v_{q+1} - v_1} \tag{34}$$

$$V_{i,q-2}^{(1,1)} = q \frac{V_{i,q-1}^{(1,0)} - V_{i,q-2}^{(1,0)}}{v_{2q-1} - v_{q-1}}. \tag{35}$$

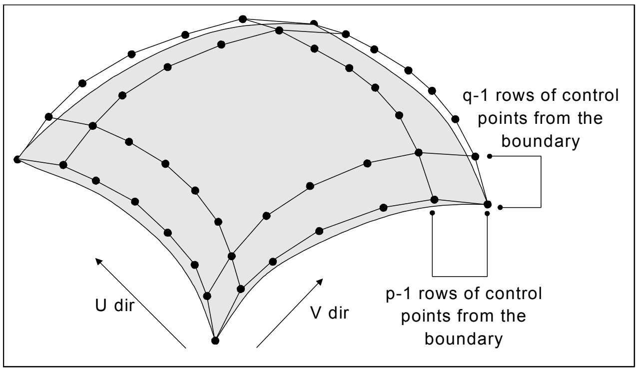

Fig. 18. Control points that should be transformed.

Figure 18 shows the control points which should not be changed to preserve C^1 continuity of a cubic B-spline surface. Thus, some detail coefficients have to be preserved if they have influence on the control points which should not be transformed. However, it is difficult to find the corresponding detail coefficients because the level of the detail coefficients is different from the level of the original surfaces as shown in (29). Also, it is difficult to add the detail coefficients to the overall sweep surface because of the difference in the level. Thus, equalizing the level of all the detail coefficients and the overall sweep surface is needed. This can be accomplished by raising the level of the detail coefficients and the overall sweep surface to the level of the original surface using the Boehm Algorithm. As described in section III.4, raising the level is the knot insertion process which results in the increase in the number of control points. Although the detail coefficients are not really of the B-spline surface, it has the same dimension as of the B-spline surface. Consequently, the knot insertion process can be applied to the detail coefficients.

In this way, we can find the corresponding detail coefficients which lies in $q-1$ arrays of control points in the u direction and $p-1$ arrays of control points in the v direction because the level of the detail coefficients and the overall sweep surface are the same as those of the original surface. Finally, after summing up all the corresponding detail coefficients to the overall sweep surface, we can get the faired surface preserving continuity with the neighboring surfaces at its boundaries.

Figure 20 shows the resultant surface faired in both directions. We can see that on the boundaries of the faired surface,

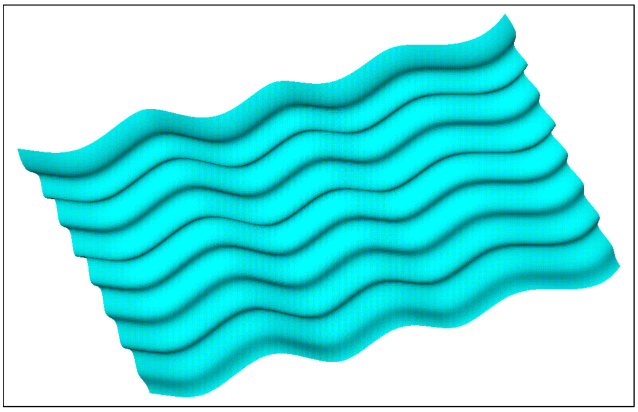

Fig. 19. Original surface.

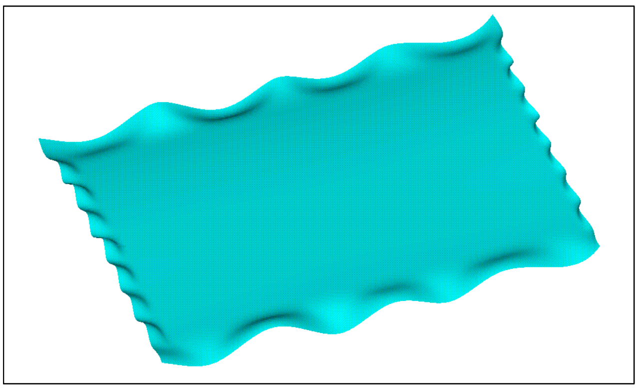

Fig. 20. Faired surface.

C^1 continuity with the original surface is preserved while the internal severe wiggles are removed.

V. CONCLUSION

Traditional methods to smooth curves or surfaces usually try to minimize the curvature variation in the curves or surfaces within a user-defined tolerance. However, they are only applicable to the correction of minute defects and not applicable to the fairing process involving very severe wiggles of high amplitude.

This paper proposes a new method for fairing such a surface or a curve using the wavelet transform while preserving the continuity with adjacent surfaces or curves. The proposed fairing method provides a similar effect on the mathematical surface to the grinding operation using sand paper on the physical surface. Besides, a user can fair the surface interactively through selecting a local region by the fractional level.

So far, this algorithm is limited to the B-spline surfaces or curves with the uniform knot vector. So, further study is needed to extend this algorithm for NURBS surfaces or curves with non-uniform knot vectors. It would be necessary to consider a new wavelet based on the vector space composed of non-uniform knots to solve this problem. Alternatively, approximation of the surface to that of uniform knot vectors would be a solution as well. Currently, the proposed algorithm is only applicable to 4 boundary B-spline patches and cannot handle trimmed surfaces. Subdividing the parametric domain of the trimmed surface into 4 boundary regions will solve this problem.

REFERENCES

[1] Forsey, D. and Bartels, R., "Hierachical B-spline Refinement," *Computer Graphics (Proceedings of SIGGRAPH '88)*, Vol. 22, No. 4, 1988, pp. 205-212.

[2] Chui, C. K., *An Introduction to Wavelets*, Academic Press, Boston, 1992.

[3] Chui, C. K. and Quak. E., "Wavelets on a Bounded Interval," *Numerical Methods of Approximation Theory*, edited by Braess, D. and Schumaker, L. L.

[4] Mallat, S., "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 11, No. 7, 1989, pp. 674-693.

[5] Chuang G. C.-H. and Kuo, C-C. J. "Wavelet Descriptor of Planar Curves: Theory and Applications," *IEEE Transactions on Image Processing*, Vol. 5, No. 1, 1996, pp. 56-70.

[6] Gortler S., "Hierarchical and Variational Geometric Modeling with Wavelets," *Symposium on Interactive 3D graphics*, edited by Hanrahan, P. and Winget, J., ACM SIGGRAPH, 1995, pp. 35-42.

[7] Farin, G. and Sapidis, N., "Curvature and the Fairness of Curves and Surfaces," *IEEE Computer Graphics and Applications*, Vol. 9, No. 2, 1989, pp. 52-57.

[8] Greiner. G., "Variational Design and Fairing of Spline Surfaces," *Computer Graphics Forum*, Vol. 3, 1994, pp. 143-154.

[9] Rando, T. and Roulier, J., "Designing Faired Parametric Surfaces," *Computer-Aided Design*, Vol. 23, No. 7, 1994, pp. 492-497.

[10] Moreton H. and Sequin, C., "Functional Optimization for Fair Surface Design," *Computer Graphics*, Vol. 26, No. 2, 1992, pp. 167-176.

[11] Finkelstein, A. and Salesin, D., "Multiresolution Curves," *Computer Graphics (Proceedings of SIGGRAPH '94)*, Vol. 28, 1994, pp. 261-268.

[12] Lounsbery, M., DeRose, T. and Warren, J., "Multiresolution Analysis for Surfaces of Arbitrary Topological Type," *ACM Transactions on Graphics*, Vol. 16, No. 1, 1997, pp. 34-73.

[13] Stollnitz, E., DeRose, T., and Salesin, D., *Wavelets for Computer Graphics: Theory and Applications*, Morgan Kaufman, San Francisco, 1996.

[14] Lee, K., *Principles of CAD/CAM/CAE Systems*, Addison Wesley, 1999.

[15] Farin, G., *Curves and Surfaces for Computer-Aided Geometric Design*, 4th ed., Academic Press, San Diego, 1996.

[16] Piegl, L. and Tiller, W., *The NURBS Book*, 2nd ed., Springer, 1997.

Joohyung Cho received his B. S. and M. S. degrees in mechanical design and production engineering from Seoul National University, Korea in 1996 and 1998, respectively. He is now a research engineer at Daewoo Heavy Industries & Machinery Ltd. since March 1998. His current research interests include computer aided geometric design, computer graphics, and signal processing.

Tae-wan Kim received his B. S. and M. S. degrees in industrial engineering from Hanyang University, Korea in 1985 and 1987, respectively and M. S. and Ph. D. degrees in computer science from Arizona State University, USA in 1993 and 1996, respectively. He worked at the Curves and Surfaces group of Structural Dynamics Research Corporation (SDRC) as a software engineer from 1996 to 1999. He then worked at National Research Laboratory for the development of solid modeling kernel at Seoul National University as a research associate from 1999 to 2001. He is an assistant professor at the department of digital contents of sejong University since March 2001. His current research interests include computer aided geometric design, computer graphics, computer games, and digital contents.

Kunwoo Lee is a professor in the school of mechanical and aerospace engineering at Seoul National University in Korea. He received his B.S.(1978) from Seoul National University, and M.S.(1981) and Ph.D.(1984) in mechanical engineering from MIT. He was previously an assistant professor in the department of mechanical & industrial engineering, University of Illinois at Urbana-Champaign, and a visiting associate professor in the department of mechanical engineering at MIT. Currently, he is on the editorial board of the journals, computer-aided design, and robotics and computer integrated manufacturing. He has been a program committee chair of Solid Modeling Symposium 2001. He is also leading the National Research Lab. on multi-resolution modeling kernel in Korea. His research interests include geometric modeling, rapid prototyping system of new concept, and application software such as injection mold design system and custom tailoring shoe system.