

NC Tool-Path Generation Based on Surfel Models Constructed from Random Scanned Data

Hong-Tzong Yau¹, Rong- Kuei Lee², Cheng-Ming Chuang³ and Chien-Yu Hsu⁴

¹National Chung-Chang University, imehty@ccu.edu.tw

^{2,3,4} National Chung-Chang University

ABSTRACT

This paper presents a new approach to the generation of NC tool-path from random scanned data. Instead of using smooth or triangulated surfaces reconstructed from the raw data, we use the point-based Surfel models to generate NC tool-paths. As a result, the tool-path generation is more efficient and still maintains the advantage of having accurate and smooth machining result. The word “surfel” is the combination of two words, “surface” and “element”. It is a newly introduced point-based surface model, originally applied to the rendering of scanned data. In this paper, the surfel model is created using an elliptical Gaussian re-sampling filter that is based on a signal re-sampling algorithm. Since the input scanned data is of discrete and random nature, we utilize the warping process to transform the input data into a continuous surface and then re-sample the continuous surface. The re-sampling filter is based on the Nyquist criterion, so that the feature of the surface will not have aliasing. Because the re-sampled data can replace the original surface, we can generate tool paths based on these point data. For cutting tools with various sizes, adaptive re-sampling schemes can be used to generate sufficient sampled points for the generation of accurate and smooth tool-paths. Compared with other tool-path generation methods using scanned data [9, 10], this new approach is based on sound signal processing principles rather than ad hoc or heuristic schemes, therefore it is more systematic and can be easily extended to multi-axis machining.

Keywords: NC machining, tool-path generation, surfel, reverse engineering.

1. INTRODUCTION

In reverse engineering, a part to be duplicated and machined is first digitized using a laser scanner and then reconstructed with triangulated surfaces or continuous surfaces (NURBS, for example). Tool paths are then calculated based on these triangulated surfaces or continuous surfaces. However, surface reconstruction is a non-trivial task that usually requires a complex reconstruction process and significant amount of computation. And if the algorithm that is used is not good enough to construct a surface without gaps, even more time will be required. To resolve these issues, this paper presents a new approach to the generation of NC tool-paths from random scanned data. Instead of using continuous or triangulated surfaces reconstructed from the raw data, we use the point-based Surfel models [1-6] to generate NC tool-paths. As a result, the tool-path generation is more efficient and still maintains the advantage of having accurate and smooth machining result.

Traditional NC tool path generation based on continuous or triangulated free-form surfaces can be divided into two major approaches. One is the generation of CL (Cutter Location) surfaces; the other is the use of z-map for creating high density of discrete surface points. The essential task is to generate interference-free tool-paths for NC machining. The tool-paths consist of a series of gouge-free CL data. The CC points approach is a point-based method, by which a CL point is found by lowering the cutter along the machining direction (z-axis, for example) until the cutter contacts with the part surface represented by the z-map data [7-11]. The CL surface approach creates the CL surface by inversely offsetting the tool surface along the original surface. The CL data are then computed by slicing the CL surface [12-17]. After the tool-paths are generated, the machining operation is to control the cutter tip to move through all CL data on the tool-paths to complete machining.

From the above description, it is clear that for both approaches to be useful, a pre-condition is the existence of the complete and continuous surface model. Although the z-map method only requires discrete data points, it still needs the original continuous surface to generate the sample points. To generate smooth and accurate NC tool-paths, the original scanned data may not have sufficient density and may contain noises. Instead of reconstructing continuous surfaces from the scanned data, the Surfel model can be utilized to overcome these problems.

The word “surfel” is the combination of two words, “surface” and “element”. It is a newly introduced point-based surface model, originally applied to the rendering of scanned data [1]. In this paper, the surfel model is created using an elliptical Gaussian re-sampling filter that is based on a signal re-sampling algorithm. Since the input scanned data is of discrete and random nature, we utilize the warping process to transform the input data into a continuous surface and then re-sample the continuous surface. The re-sampling filter is based on the Nyquist criterion, so that the feature of the surface will not have aliasing. Because the re-sampled data can replace the original surface, we can generate tool paths based on these point data. For cutting tools with various sizes, adaptive re-sampling schemes can be used to generate sufficient sampled points for the generation of accurate and smooth tool-paths. This is a new method and is different from the traditional methods of tool path generation that are based on continuous surfaces or triangulated surfaces.

The rest of the paper is organized as two major parts. The first part describes the method used to construct a surfel model and to re-sample the surface. The second part explains the algorithm used for 3-axis projection machining. Cutter location points are generated similar to the z-map approach, except they are now generated on the surfel model. To search the re-sample points created from the surfel model, the quad-tree data structure is adopted for efficient query search. We then evaluate the step error with a simple and efficient method. New CL points are then interpolated and inserted using a binary tree. Simulations and experiments show that this proposed method can produce precision NC tool-paths from random scanned data with efficiency.

2. SURFEL MODEL

Before introducing the surfel model, we will introduce the concept of re-sampling. Re-sampling includes four steps:

- (1) Constructing a continuous signal from a discrete signal.
- (2) Warping the domain of the continuous signal.
- (3) Warping and filtering the continuous signal using a pre-filter.
- (4) Sampling the continuous signal to produce the discrete output signal.

Let the input signal be $f(U)$ and the forward mapping be $X = m(U)$,

where $U = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}$ is the source space and $X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$ is the screen space. The base function is $r(U)$. The pre-

filter is $h(U)$, and the output signal is $g(X)$.

The equations for re-sampling are as follows:

- (1) The equation for constructing a continuous signal from a discrete signal:

$$f_c(U) = f(U) \Theta r(U) = \sum_{K \in S^n} f(K) r(U - K) \quad (1)$$

S^n : source coordinates

- (2) The equation for warping to projection space:

$$g_c(X) = f_c(m^{-1}(X)) \quad (2)$$

- (3) The equation for projection space signaling using a pre-filter:

$$g_c'(X) = g_c(X) \Theta h(X) = \int_{D^n} g_c(T) h(X - T) dT \quad (3)$$

D^n : destination coordinates

- (4) The equation for sampling the output signal:

$$g(X) = g_c'(X) i(X) \quad (4)$$

where Θ denotes a convolution.

After defining the equations of the re-sampling process, we can get the equation of the re-sampling filter by expanding the above relationships in reverse order.

$$g(X) = g'_c(X)i(X) = \left[\sum_{k \in S^n} f(K)\rho(X, K) \right] i(X) \tag{5}$$

where $\rho(X, K) = \int_{D^n} h(X - T)r(m^{-1}(T) - K)dT$ is the ideal re-sampling filter.

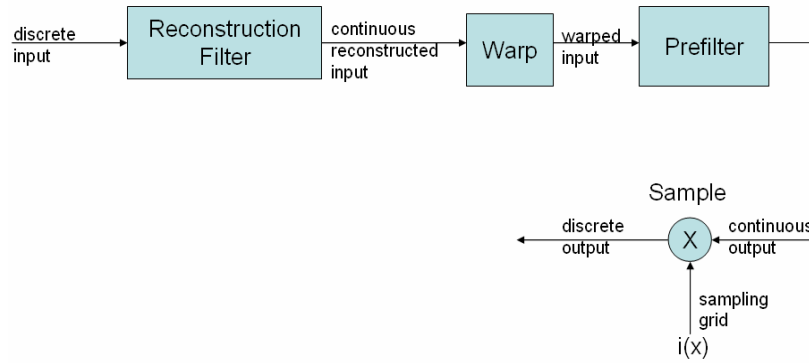


Fig. 1. A block diagram of the ideal re-sampling filter.

Previously, we constructed a reconstruction filter and a pre-filter using an ideal filter. Here we combine a reconstruction filter and a pre-filter as an elliptical Gaussian low-pass filter to get the elliptical Gaussian re-sampling filter. The equation for the elliptical Gaussian low-pass filter is as follows:

$$g_V(X) = \frac{1}{2\pi|V|^{1/2}} e^{-\frac{1}{2}XV^{-1}X^T} \tag{6}$$

where $V^{-1} = \begin{bmatrix} A & B/2 \\ B/2 & C \end{bmatrix}$

The equation for the ellipse is: $Ax^2 + Bxy + Cy^2 = F$ (7)

We transform the reconstruction filter $r(U)$ and the pre-filter $h(U)$ to the elliptical Gaussian low-pass filter $r(U) = g_{V_r}(U)$ and $h(U) = g_{V_h}(U)$. Then we can get an equation for the elliptical Gaussian re-sampling filter, which is the equation for the Surfel model. [4]

$$\rho'(U) = h'(U)\Theta r(U) = g_{J^{-1^T}V_h J^{-1}}(U)\Theta g_{V_r}(U) = g_{J^{-1^T}V_h J^{-1} + V_r}(U) \tag{8}$$

Considering the input signal $f(U)$ and the output signal $g(X)$, we have

$$g(X) = \left[\sum f(K)g_{J^{-1^T}V_h J^{-1} + V_r}(m^{-1}(X) - K) \right] i(X) \tag{9}$$

The filter is in source space, and there is another elliptical Gaussian re-sampling filter in projection space. The implementation of the Surfel model in this paper is according to the elliptical Gaussian re-sampling filter in projection space. [1-4]

$$g(X) = \left[\sum f(K)g_{V_h + J^T V_r J}(X - m(K)) \right] i(X) \tag{10}$$

Actually, an elliptical Gaussian re-sampling filter is a blending function. Each input signal $f(K)$ has a corresponding weighting function $\frac{g_{I+J^T IJ}(X-m(K))}{\sum g_{I+J^T IJ}(X-m(K))}$. We perform interpolation according to the space relation of the blending function.

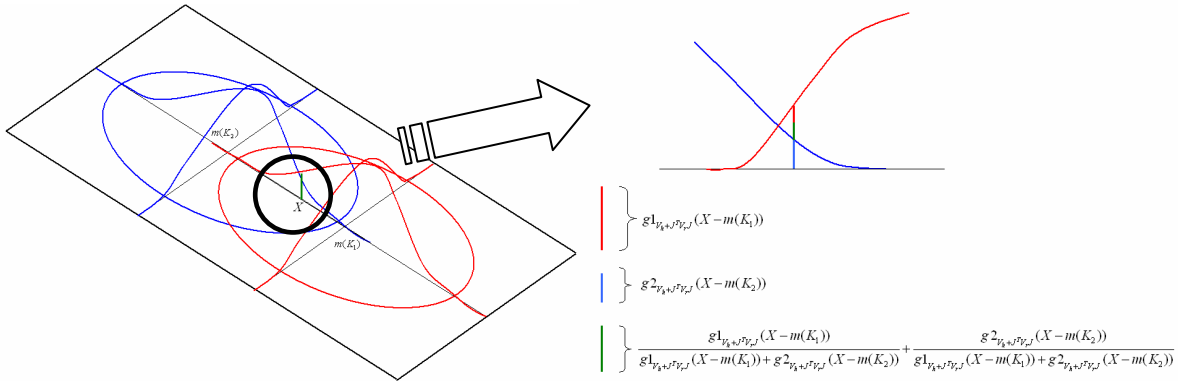


Fig. 3. The physical meaning of the average weight.

4. TOOL-PATH GENERATION FOR 3-AXIS PROJECTION MACHINING

The Surfel model provides an efficient and powerful way to resample the workpiece in the projection space. In the projection space, a “Z-map” representation of the surface can be constructed for the calculation of cutter locations and tool-paths. The projection direction is parallel to the cutting direction, which can be either of the 3 Cartesian axes, or any of the tilted tool direction in multi-axis machining. Using the cutter center as the origin, we can deploy an entire x-y projection plane for 3-axis machining. For multi-axis machining, the x-y projection plane can be set up slightly larger than cutter size. We then project the x and y coordinates on the part surface to get the corresponding z coordinates. The cutter is then lowered in the z-direction from the pre-set height until it touches the part surface. The contact will take place at the cutter contact point (CC point), it will be used to find the cutter location point (CL point, also the C point in Fig. 4). The region of the part to be machined on which the cutter is projected is called the cutter contact region (CC region). In Figure 4, point r_{cc} is the CC point, point r_c is the CL point, and the area inside the circle is the CC region.

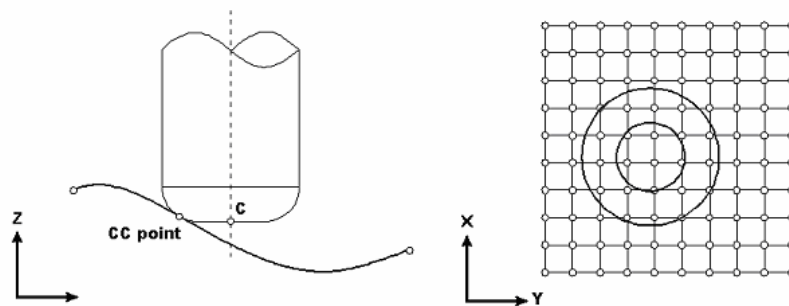


Fig. 4. CC point, CL point and CC region

The algorithm of tool path generation for projection machining follows the traditional Z-map approach by finding the “lowest” cutter position which will not cause interference between the cutter and the part surface. Therefore, for each sampled point in the projection space that falls inside the CC region, the z value of the CL point is calculated assuming it is a CC point. The obtained maximum z value, together with the x and y values of the cutter center position, are then

used as the coordinates of the CL point. In the following, we outline the equations used to determine the CL points for various types of cutters that are often used.

Flat-end cutter:

Assume that the vertex point is $P = (X_p, Y_p, Z_p)$ and that the radius of the cutter is R . The center point of the bottom of the cutter is $C = (X_c, Y_c, Z_c)$.

- (1) First, check whether the point P is inside the CC region:
If $\left((X_p - X_c)^2 + (Y_p - Y_c)^2 \right)^{1/2} \leq R$, point P is inside the CC region.
- (2) For points that are inside the CC region, Z_p is the maximum value of Z_c .

Ball-end cutter:

Assume that the vertex point is $P = (X_p, Y_p, Z_p)$ and that the radius of the cutter is R . The center point of the bottom of the cutter is $C = (X_c, Y_c, Z_c)$.

- (1) First, check whether the point P is inside the CC region:
If $\left((X_p - X_c)^2 + (Y_p - Y_c)^2 \right)^{1/2} \leq R$, point P is inside the CC region.
- (2) Calculate the maximum value of Z_c according to the points in the CC region:

$$Z_c = Z_p + \left(R^2 - (X_c - X_p)^2 - (Y_c - Y_p)^2 \right)^{1/2} \quad (13)$$

Fillet-end cutter:

Assume that the vertex point is $P = (X_p, Y_p, Z_p)$ and that the body radius of the cutter is R . The corner radius is r , and the center point of the bottom of cutter is $C = (X_c, Y_c, Z_c)$.

- (1) First, check whether the point P is inside the CC region:
If $q = \left((X_p - X_c)^2 + (Y_p - Y_c)^2 \right)^{1/2} \leq R$, point P is inside the CC region.
- (2) Calculate the maximum value of Z_c according to the points in the CC region: (14)

The above method helps us to find the CL point for each specified cutter position in the projection plane. However, to control the machining accuracy, step error (cordal deviation) and interval error (scallop height) need to be controlled within the allowable tolerance, which will be discussed in the next section.

5. CONTROL OF MACHINING ACCURACY

We discussed the method of calculating the CL point of the cutter in the previous section. The tool-path is then composed of the serial connections of these CL points. Tool-path planning is based on two user-defined parameters: step length and path interval. The step length is the length between two neighboring CL points along the cutter moving direction. The path interval is the length between two neighboring cutter paths. Default values will be assigned to these two parameters (e.g., 0.5 mm), but additional CL points will be inserted if step or interval error exceeds the machining tolerance (e.g., 0.01 mm). In this paper, we propose an efficient and effective method to determine the step error. As shown in Fig. 5, the CL and CC points can be used to determine the tangent directions of machined surface. Using the intersection of the tangent lines, we can approximately estimate the step error \mathcal{E} by the projection of the intersection point to the CC connection line. If \mathcal{E} is larger than the machining tolerance, a new CL point will be inserted at the intersection point. This process is recursively repeated until the machining tolerance is satisfied. Similar to the convex hull property of control points, the approximation error will eventually converge to the real step error.

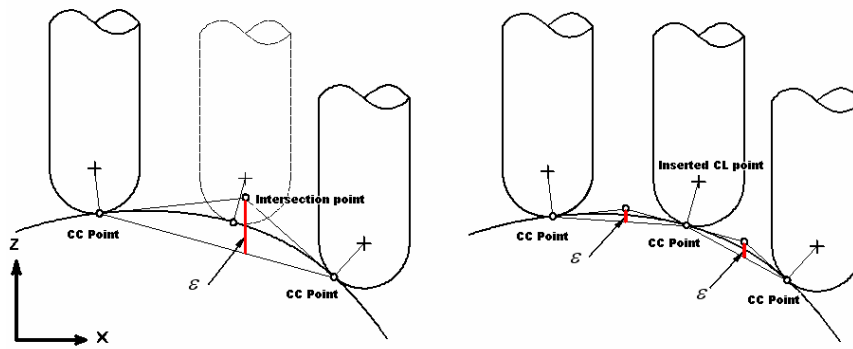


Fig. 5. CL point insertion based on estimated step error

As new CL points are interpolated to reduce the tool path error, a binary tree is used to keep track of new CL points interpolated between two CL points. Fig.6 shows the recursive CL point insertion in the cutter moving direction to reduce the step error. Likewise, a new cutter path can be inserted between two neighboring cutter paths if the maximum interval error is larger than the machining tolerance. In this work, we use the model given by Lin and Koren [18] to calculate the interval error.

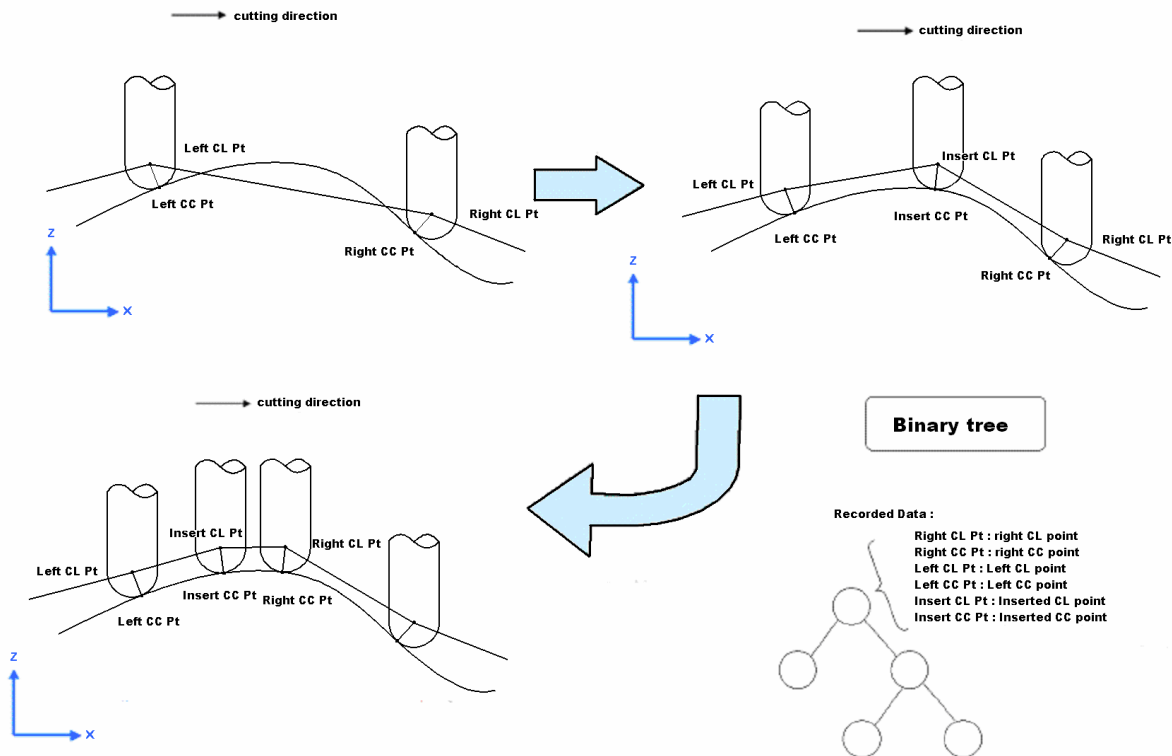


Fig. 6. Recursive CL point insertion and its binary tree

Adaptive Re-Sampling Schemes

For cutting tools with various sizes, adaptive re-sampling schemes can be used to generate sufficient sampled points for the generation of accurate and smooth tool-paths. This is really one of the advantages of the proposed approach. By maintaining the required machining accuracy, smaller size sampled data can be generated for larger cutting tools. In figure 7, assuming the grid size of the projection plane is d and the radius of the cutter is R , we can estimate the cave-in distance h . By limiting h to a small tolerance, say, 5 micron, we can evaluate the necessary grid size d as Eq. (15).

$$h = R - \sqrt{R^2 - (d/2)^2}$$

$$d = \sqrt{8h(R-h)} \quad (15)$$

For example, if h is 0.005 mm and R is 5 mm, the grid size is 0.45 mm. But if R is 20 mm, the grid size d becomes 0.89 mm, which means the sampling size can be reduced to only one quarter.

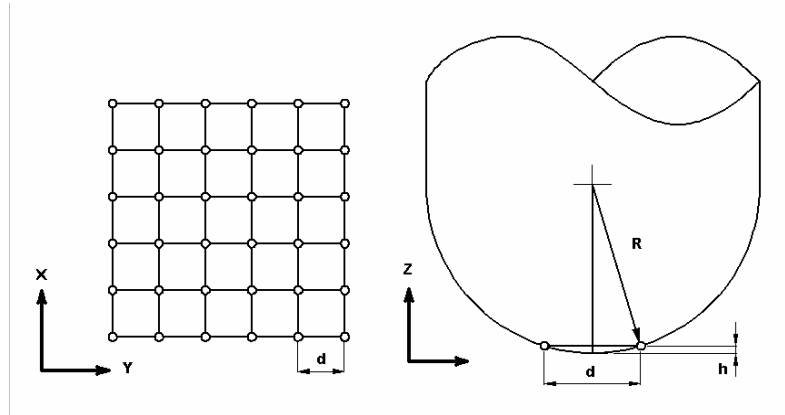


Fig. 7. Adaptive re-sampling scheme based on cutter size

6. EXAMPLES

As the first example, a toy air-plane was scanned to produce the random scanned data for testing. Fig. 8(a) shows the scanned data obtained by a digitizer. The number of the scanned points is 15,072. Then, the data is re-sampled in the projection space (Fig. 8(b)) and warped from the projection space to the source space (Fig. 8(c)). The number of the re-sampled points is 137,945. The tool-paths generated in x and y direction are shown in Fig. 8(d) and Fig. 8(e) respectively.

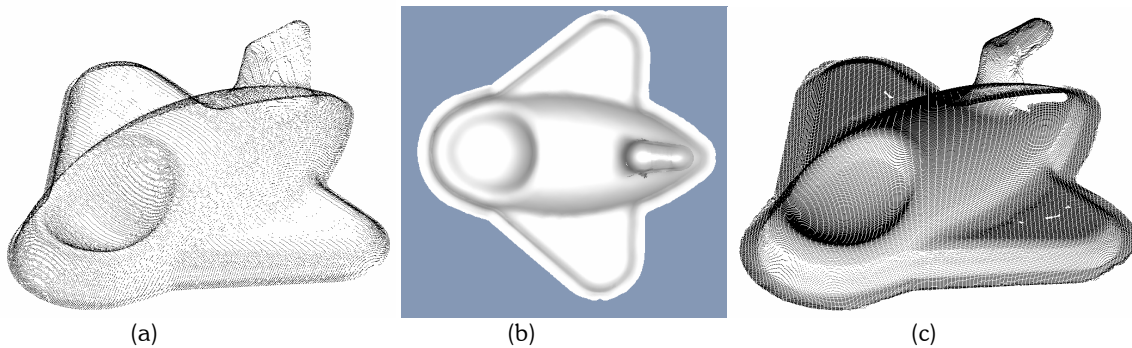


Fig. 8. (a)The random scanned data. (b)Re-sample the surface in projection space. (c) Warp the projection space to source space.

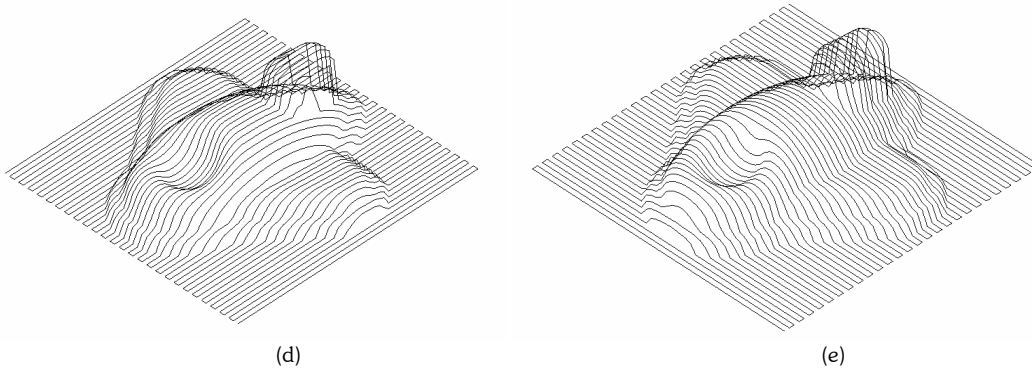


Fig. 8. (d) Machining along the x-direction. (e) Machining along the y-direction.

The second example is from the scanning of the wood pattern of a shoe sole. The model can be machined using a 3-axis, 5-side machining approach on a 3-axis machining center equipped with an indexed rotary table. Fig. 9 shows the five different projection spaces and the corresponding re-sampled data using the proposed approach. This example shows that the proposed method can be extended to tool-path for 5-axis machining as long as the re-sampled data is generated in the projection plane normal to the tool axis direction.

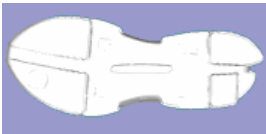
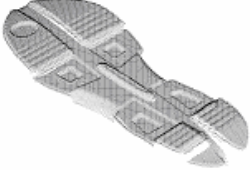



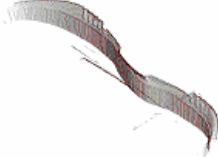

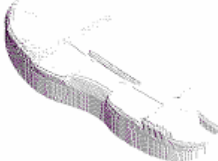


View-port	Re-sampled data in projection space (2D)	Re-sampled data in source space(3D)
Top view		
Front view		
Right view		
Left view		
Back view		

Fig. 9. Multiple views of re-sampled data for a shoe model

7. CONCLUSION

In this paper, a new tool-path generation method is proposed for NC machining of random scanned data. Instead of using ad hoc or heuristic rules to preprocess the scanned data, this paper proposes the use of Surfel model for generating NC tool-paths. The surfel model is created using an elliptical Gaussian re-sampling filter that is based on a signal re-sampling algorithm. Since the input scanned data is of discrete and random nature, we utilize a warping process to transform the input data into a continuous surface and then re-sample the continuous surface. For cutting tools with various sizes, an adaptive re-sampling scheme is developed to generate sufficient sampled points for the calculation of accurate and smooth tool-paths in the tool projection space. The CL point is then found by lowering the cutter in the direction normal to the projection plane when contacts with the re-sampled points are made. To control the required machining accuracy, a simple yet efficient CL point insertion method is proposed to recursively add new CL points between neighboring CL points when the step error exceeds the machining tolerance. The proposed approach can potentially be extended to multi-axis machining by re-sampling the scanned data in the projection plane normal to the cutter tool axis in 5-axis orientations.

8. REFERENCES

- [1] Pfister, H., Zwicker, M., Baar, J. V. and Gross, M., "Surfels: Surface Elements as Rendering Primitives" *Proceedings of SIGGRAPH 2000*, pp 335-342.
- [2] Matthias Zwicker, Hanspeter Pfister, Jeroen van Baar, Markus Gross, "Surface Splatting" , *Proceedings of SIGGRAPH 2001*, pp 371-378.
- [3] Ren, L., Pfister, H. and Zwicker, M., Object Space EWA Surface Splatting: A Hardware Accelerated Approach to High Quality Point Rendering, *EUROGRAPHICS 2002*, pp 461-470.
- [4] Zwicker, M., Pauly, M., Knoll, O. and Gross, M., Pointshop 3D: An Interactive System for Point-Based Surface Editing, *Proceedings of SIGGRAPH 2002*, pp 322-329.
- [5] Zwicker, M., Pfister, H., Baar, J., and Gross M., EWA Splatting, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 8, No. 3, 2002, pp 223-238.
- [6] Zwicker, M., Pfister, H., Baar, J., and Gross M., EWA Volume Splatting, *Proceedings of IEEE Visualization 2001*, pp 29-36.
- [7] Hwang, J. S. and Chang, T. C., Three-axis machining of compound surfaces using flat and filleted endmills, *Computer-Aided Design*, Vol. 30, 1998, pp 641-647.
- [8] Hwang, J. S., Interference-free tool-path generation in the NC machining of parametric compound surfaces, *Computer-Aided Design*, Vol. 24, 1992, pp 667-676.
- [9] Choi, B.K. and Jerard, R. B., *Sculptured Surface Machining*, Kluwer Academic Publishers, 1998.
- [10] Lin, A. C. and Liu, H. T., Automatic generation of NC cutter path from massive data points, *Computer-Aided Design*, Vol. 30, 1998, pp 77-90.
- [11] Park, S. C., Tool-path generation for Z-constant contour machining, *Computer-Aided Design*, Vol. 35, 2003, pp 27-36.
- [12] Chuang, C. M., *The tool path generation algorithm of STL models for NC machining and its application*, Ph.D. Thesis, National Chung-Chang University, Taiwan, 2004.
- [13] Yau, H. T., Chuang, C.M. and Lee, Y.S., Numerical control machining of triangulated sculptured surfaces in a stereo lithography format with a generalized cutter, *International Journal of Production Research*, Vol. 42, No. 13, 2004, pp 2573-2598.
- [14] Park, S. C., Sculptured surface machining using triangular mesh slicing, *Computer-Aided Design*, Vol. 36, No. 2, 2004, pp 279-288.
- [15] Tang, K., Cheng, C.C. and Dayan, Y., Offsetting surface boundaries and 3-axis gouge-free surface machining, *Computer-Aided Design*, Vol. 27, No. 12, 1995, pp 915-927.
- [16] Choi, B. K., Kim, D. H. and Jerard, R. B., C-space approach to tool-path generation for die and mold machining, *Computer-Aided Design*, Vol. 29, No.9, 1997, pp 657-669.
- [17] Inui, M., Fast Inverse Offset Computation using polygon rendering hardware, *Computer-Aided Design*, Vol. 35, No. 2, 2003, pp 191-201.
- [18] Lin, R. and Koren, Y., Efficient Tool-Path Planning for Machining Free-Form Surfaces, *ASME Journal of Engineering for Industry*, Vol. 188, No.1, 1996, pp 20-28.