日本バーチャルリアリティ学会第8回大会論文集(2003年9月)

プロダクトデザイナーの為のデジタルデザインスケッチ

Digital Design Sketch for Product Designers

錫亨[1], 木島竜吾[2]
Seok-Hyung BAE and Ryugo KIJIMA

1) 岐阜大学 バーチャルシステムラボラトリー
(〒501-1193 岐阜県岐阜市柳戸 1-1, bae@vsl.gifu-u.ac.jp
2) 岐阜大学 工学部
(〒501-1193 1193 岐阜県岐阜市柳戸 1-1, kijima@info.gifu-u.ac.jp

Abstract : So far the effort of digitalizing product-styling tools for the early design development stage has mainly focused on using raster-type graphics S/Ws. However, it has become a new obstacle to concurrent engineering because resultants of these S/Ws are no more than images, which can be hardly converted to digital forms required by downstream process. For resolving this problem, this paper suggests the digital design sketch based on the observation and analysis of design-sketch elements and behaviors. The proposed method not only allows designers to perform their creative work sensually by adopting user-friendly interface based on their traditional design sketch, but also provides a digital connectivity with downstream process by handling strokes as vector-type data.

Key Words: *digital design sketch, digital connectivity, vector-type data handling*

1. Introduction

Digitalizing has become a general tendency in every field of the world. As industries make efforts to digitalize all the product-development process in view of *concurrent engineering* , the design and development process is also changing dramatically.

Focusing on *product styling* with special emphasis on external shapes, the digitalizing effort can be categorized into two. For the early design phase consists of the *concept selection* and *embodiment design*, *raster-type* 2D graphics S/Ws are used. For the *detailed design*, RE (reverse engineering) and CAD S/Ws are used.

However, the digitalization for the early-phase design development is somewhat controversial especially from a concurrent-engineering point of view. Undoubtedly, raster-type 2D graphics S/Ws are quite compatible with the designer's work in the concept selection and embodiment design stages (*rapid*, *flexible*, *sensuous*). But, raster-type tools are basically *image-based*, and inevitably suffer serious problems in *touch editing* and *history storing*. Furthermore, they have *no connectivity* to downstream *digital* processes. Thus, the styling alternatives in the raster-image form must be generated again with CAD S/Ws, and it unavoidably introduces not only numerical (or dimensional) errors but also the unwanted change of the feeling (or impression) of styles.

Therefore, this paper proposes a new digital-styling tool so that designers can express their ideas naturally while providing a digital connectivity. In order to identify the *design sketch* , we used several information gathering methods: 1) a *simple observation* of numerous design-sketch works, 2) *disposable-camera* technique to gather designers' working style, and 3) in-depth *interviews* with professional industrial-designers. Based on the design-sketch elements and behavior patterns extracted from the factor analysis, digital functionalities, are mapped with them under proper assumptions and the *vector-type data handling scheme.*

This paper is organized as the following order: after related work summarized in Section 2, a design sketch is considered as the designer's versatile tool in Section 3. Based on the analysis of design sketch in Section 4, the solution mapping for the digital-sketch emulation is given in Section 5. The user's experience and feedback about a prototype program are shown in Section 6 followed discussions and conclusions in Section 7 and Section 8, respectively.

2. Related Work

The pen-based interface has been a common human-computer interface. It can highly increase the level of intimacy when combined with 2D graphics tools [6][12] because of their inherent analogy. Many studies tried to use freehand sketch for CAD modeling [5][9][11][13], but most of them focus on feature recognition of some primitives. On the other hand, there were little studies about sketch stroke itself: Baudel [3] suggested the *mark-based spline modification* where an initial spline can be updated in real-time by the user's successive strokes. Arvo and Novins [1] proposed so called the *fluid sketch* based on the continuous recognition of freehand sketches and optimal-shape morphing. Roth-Koch [10] made overtures of a off-line *B-spline fuzzy array* concept.

3. Why Design Sketch

In virtue of distinctive advantages of sketch as follows: 1) the *simplicity* of its way and equipment, 2) the *speed*, 3) the *easiness* to catch ideas, etc., a sketch has been widely used for creative works. Especially in real product styling, so called the design sketch is popular for effectively describing the 3D shapes of product: 1) the *concept description* with texts, 2) the *shape description* with color, texture, and shading effects, 3) the *layout description* with a scale, and 4) the *final product description* with full details. As the sketch is versatile, the advantages of digitalizing it are many as: 1) the almost merits of a physical sketch can be inherited, 2) some weak points–the inaccuracy problem caused by delicate hand shaking, imperfect proportion or perspective, annoying cleaning up useless lines in the end–can be complemented, 3) the problems of storage, retrieval, and modification of physical sketches can be settled or improved, and 4) designers can easily fit the new digital tool.

4. Analysis of Design Sketch

To digitalize design sketch, we conducted a wide range of observations for identifying how designers make design sketches, and how their working environment looks like. We collected numerous design-sketch works, videotaped professional industrial-designers, and interviewed them. Throughout careful observations on enormous materials, we extract several patterns on design-sketch elements and behaviors.

Proposition 1 *There are two design-sketch elements: 1) a smooth curve without an inflection point, and 2) an arbitrarily rotated ellipse.*

Proposition 2 *There are two design-sketch support elements: 1) a straight line, and 2) an exact circle.*

Following are the several propositions about design-sketch behaviors:

Proposition 3 *Drawing* dimensionally *accurate curves is primarily difficult.*

Proposition 4 *The designer's sketching process is a kind of adaptation with touch repetition based on visual feedback.*

Proposition 5 *When drawing a new curve, the existing marks of previous scribbles function as guides.*

Proposition 6 *The clean-up process is an essential step.*

5. Design-Sketch Emulation in Digital Form

5.1 Design-Sketch Element Mapping

Shown in Fig. 1 is the solution mapping related to design-sketch elements. A smooth curve without an inflection point is interpreted as *simply tensioned-smooth spline*, and mapped to a *cubic* Bezier curve. An ellipses is mapped to a quadratic NURBS curve composed of four rational-quadratic Bezier curves. The two support elements, a straight line and exact circle, are mapped to a linear Bezier curve and a quadratic NURBS curve, respectively.

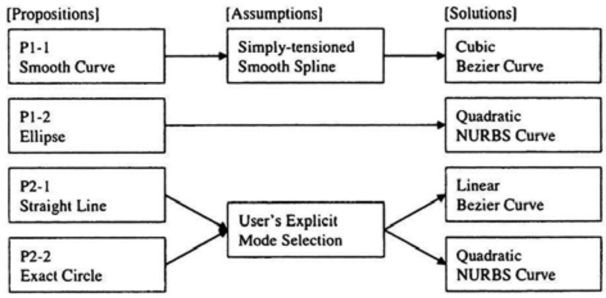

Fig. 1: Design-sketch element mapping

5.2 Design-Sketch Behavior Mapping

In this section, the analyzed sketch behaviors are synthesized into the design-sketch behavior map as shown in Fig. 2.

Whenever the pen-point changes, the trajectory of a scribble is approximated to the corresponding mathematical (NURBS) model. The approximation somewhat compensate the dimensional inaccuracy of human drawing by eliminating delicate waves caused by hand shaking. The proposed method enables designers to create a satisfactory curve in their mind by allowing repetitive scribbling. When the designer decides if overlapped scribbles form an embodied shape enough to shrink a final curve, the settlement is executed with his/her explicit order.

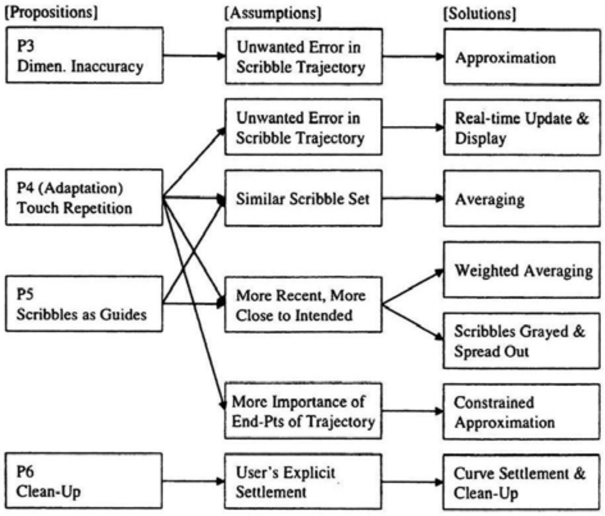

Fig. 2: Design-sketch behavior mapping

We assumed that more recent scribble is closer to an intended curve than old ones because the designer can adapt his/her drawing with visual feedback. Hence, the settled curve, $\bar{\mathbf{c}}(t)$, is a *weighted average* of a set of scribble curves,

$\{\mathbf{c}_i | i = 0, \ldots, s-1\}$, given as:

$$\bar{\mathbf{c}}(t) = \sum_{i=0}^{s-1} w_i \mathbf{c}_i(t) \qquad (1)$$

where $w_i = \frac{b^i(b-1)}{b^s-1}$; $b(\geq 1)$ is the ratio of weights of two consecutive levels. The designer is able to expect the resultant curve before settlement using information provided in the forms of scribbles' display attributes, *darkness* and *width* –as the number of scribbles increases, the previous scribbles are grayed and spread out. The gray level of the i^{th} scribble, $g[i]$, is calculated as:

$$g[i] = (g_{max} - g_{-\infty})(b_g^{i-s+1} - 1) + g_{max} \qquad (2)$$

where g_{max} is for the latest (or current) scribble, $g_{-\infty}$ is the lower-bounded gray level ($g[i] \rightarrow g_{-\infty}$ as $i \rightarrow -\infty$), and $b_g(\geq 1)$ is the ratio of two consecutive gray levels. Similarly, the equations for the width of the i^{th} scribble is given as follows:

$$z[i] = (z_{-\infty} - z_{min})(1 - b_z^{i-s+1}) + z_{min}. \qquad (3)$$

Depicted in Fig. 3 are illustrative examples of sketching a smooth curve and ellipse.

5.3 Implementation

A prototype is implemented as a *Java Applet* program using Java^{TM} 2 Platform Standard Edition (J2SE^{TM}), Java 3D^{TM} API 1.3, and JAMA (Java Matrix Package) [7]. The main functionalities are focused on only drawing four design elements: The user can sketch curves using

Fig. 3: Sketching simple smooth curve and ellipse

one hand, and input explicit orders (key-in) using the other hand simultaneously such as the change of design-sketch elements, the settlement of a curve, the unwanted scribble removal, the initialization of the program, etc. We tested the program with a commercial tablet (WACOM Intuos^{TM}2 9″ × 12″) for a pen-based interface.

6. User Experience

Many professional industrial designers and graduate students tested our prototype. All of them became very skillful to use the program only with several trials (see Fig. 4). Especially they were favorable in following points of view: 1) an intuitive way to create curves with their adaptation, 2) an intimate drawing guide with pre-drawn scribbles' darkness and width, 3) a natural way to create an ellipse (see Fig. 3), 4) a smooth nature of resultant curves, etc.

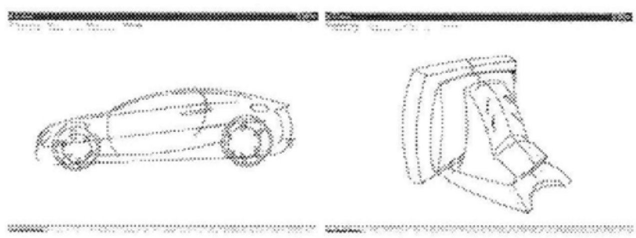

Fig. 4: Selected design-sketches using prototype

They remarked on many features for our program's improvement, which are listed up as below:

- Some wanted our program to allow alternating of the start and end point for a smooth curve, and both clockwise and counterclockwise scribbling for an ellipse.
- Many pointed out that touch pressure is also important for natural sketching.
- A few users commented an unnecessary explicit settlement when a new scribble is quite different from previous ones
- Some suggested not to throw away the scribbles after curve settlement for using them as guides for

another shape.

7. Discussions

Our strategy of *vectorizing* a scribble trajectory is a kind of *approximation*. Unlike our concern that the approximated curve might be differ from the designer's original intention, rather, they liked the smooth appearance of resultant curves.

We chose a *cubic* Bezier curve as the model curve for a smooth curve, and it looks proper so far because there were no complaints about the lack of the degrees-of-freedom among product designers. However, for finding the *optimal* degree, a quantitative study will be followed.

Although we analyzed that one of the major design-sketch elements is a smooth curve *without* an inflection point in Section 4, we did not impose any actual convexity constraint when approximating a scribble trajectory to a cubic Bezier curve. For the strong convexity constraint, it is inevitable to deal with a NLP problem [8].

8. Conclusions

In this paper, a new digital-styling tool for the designer's creative work, was suggested by emulating a physical design-sketch in computer environment. In order to identify the process of designers' sketch, we performed several information gathering methods including the simple observation of numerous designers' sketch works, disposable-camera techniques, and in-depth interviews. Based on the factor analysis of a design sketch, design-sketch elements and design-sketch behaviors are extracted, and mapped to digital functionalities for developing vector-type graphics S/Ws. The proposed method was implemented as a simple Java Applet program, and set a high value by a professional industrial-designer group.

By digitalizing a design sketch, while the merits of a physical sketch are wholly inherited, the demerits of physical sketch (for example, storage, retrieval, edit, etc.) can be complemented. This new digital media can easily be for the designer's actual work without serious resistance. Because our digital-styling tool was designed on vector-type data treatment scheme, it can resolve the many problems raster-type graphics S/Ws have. Most of all, it can provide a digital connectivity with the downstream process, and allow concurrent engineering throughout the whole product-development process.

The proposed digital-sketch concept has a lot of possibilities to be extended as follows: 1) a new form of regular vector-type 2D graphics S/W, 2) a plug-in form of existing vector-type graphics or CAD S/Ws as a 2D input device, 3) an alternative user-interface method for most sketch-based applications, 4) a direct 3D freeform-shape creation S/W for design professionals [2].

Acknowledgements

We sincerely thank a lot of designers and the graduate students of Department of Industrial Design, KAIST, who tested our program and gave valuable comments. Especially, we express our gratitude to the designers of Team Design, Design Dream, who permitted us to videotape their sketch work.

References

[1] Arvo, J., Novins, K.: Fluid Sketches: Continuous Recognition and Morphing of Simple Hand-Drawn Shapes. The Proceedings of UIST (2000)

[2] Bae, S.-H., Kijima, R., Kim, W.-S.: Digital Styling for Designers: 3D Plane-Symmetric Freeform Curve Creation Using Sketch Interface. LNCS 2669 (2003) 701-710

[3] Baudel, T.: A Mark-Based Interaction Paradigm for Free-Hand Drawing. The Proceedings of UIST (1994)

[4] Doblin, J.: Perspective: A New System for Designers. Whitney Publications, NY (1956)

[5] Eggli, L., Hsu, C.-Y., Bruderlin, B.D., Elber, G.: Inferring 3D Models from Freehand Sketches and Constraints. CAD, 29(2) (1997) 101-122

[6] Igarashi, T., Matsuoka, S., Kawachiya, S., Tanaka, H.: Interactive Beautification: A Technique for Rapid Geometric Design. The Proceedings of UIST (1997)

[7] http://math.nist.gov/javanumerics/jama/

[8] Pigounakis, K.G., Kaklis, P.D.: Convexity-Preserving Fairing. CAD 28(12) (1996) 981-994

[9] Qin S., Wright D., Jordanov I.: From On-line Sketching to 2D and 3D Geometry: A System Based on Fuzzy Knowledge. CAD 32(14) (2000) 851-866

[10] Roth-Koch, S.: Generating CAD Model from Sketches. The Proceedings of Workshop on Geometric Modelling (2000)

[11] Schweikardt, E., Gross, M.D.: Digital clay: Deriving Digital Models from Freehand Sketches. The Proceedings of CHI (2002)

[12] Tolba, O., Dorsey, J., McMillan, L.: A Projective Drawing System. The Proceedings of Symposium on Interactive 3D Graphics (2001)

[13] Zeleznik, R.C., Herdon, K.P., Hughes, J.F.: SKETCH: An Interface for Sketching 3D Scenes. The Proceedings of SIGGRAPH (1996)