

Digital Styling for Designers: Sketch Emulation in Computer Environment

Seok-Hyung Bae^{1,3}, Won-Sup Kim², and Eun-Sook Kwon²

¹ Virtual Reality Research Center, Korea Advanced Institute of Science and Technology, 373-1 Guseong-dong, Yuseong-gu, Daejeon, 305-701, Republic of Korea
bae@vr.kaist.ac.kr

<http://vr.kaist.ac.kr/~bae>

² Department of Industrial Design, Korea Advanced Institute of Science and Technology, 373-1 Guseong-dong, Yuseong-gu, Daejeon, 305-701, Republic of Korea
{iron,eskwon}@mail.kaist.ac.kr

³ Virtual System Laboratory, Gifu University, 1-1 Yanagido, Gifu, 501-1193, Japan
bae@vsl.gifu-u.ac.jp

Abstract. The digitalization of product-styling tools for the early design-development stage has focused on using raster-type graphics S/Ws. These kinds of S/Ws enable designers freely to generate ideas, and enhance them based on their creativity and aesthetic senses. However, resultants from raster-type tools are no more than images, which are hardly converted to digital forms required by downstream processes. Thus, it still remains as an obstacle for both designers and engineers to work harmoniously and simultaneously in the new-product development process through concurrent engineering. This paper proposes a new digital-styling tool for designers by emulating a design sketch. The research process is summarized as: 1) observing designers' sketch behaviors and their works, 2) analyzing designers' sketch behavior patterns for extracting sketch elements, and 3) mapping them to digital forms. The proposed digital-sketch method not only allows designers to perform their creative work, but also provides digital connectivity for concurrent engineering.

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 with a keyword, *dimension*. For the early phase consists of the *concept selection* and *embodiment design*, where idea generation and hybridization are repeated until a satisfactory styling concept is built up, *raster-type* 2D graphics S/Ws are used. For the *detailed design* where every component specification and technical drawings are made, 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: 1) the *rapid* generation of various design alternatives, 2) the *flexible* hybridization of previous ideas, 3) the *sensuous* working style, etc. But, raster-type tools are basically *image-based*, and inevitably suffer serious problems as follows: 1) the difficulty of *touch editing*, 2) the hardship of keeping a *touch history*, 3) no *connectivity* to downstream *digital* processes, etc.

Above all, no digital connectivity is the acutest problem for the harmonious concurrent engineering on the whole product-development process. The styling alternatives in the raster-image form must be converted to computer models with CAD S/Ws, separately. Transforming design works into CAD models unavoidably introduces not only numerical (or dimensional) errors but also the unwanted change of the feeling (or impression) of styles. Furthermore, when receiving feedback from downstream processes, the partial modification can hardly be done.

Therefore, this paper proposes a new digital-styling tool for designers being able to express their ideas naturally by emulating a *design sketch* in computer environment while keeping a digital connectivity. In order to identify the designer's sketch, we used several information gathering methods: 1) a *simple observation* of numerous design-sketch works, 2) *disposable-camera* technique to gather designers' working environment, 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 (or solutions), are mapped with proper assumptions and the *vector-type data handling scheme*. The proposed method not only allows designers to perform their creative work sensually with it by adopting user-friendly interface based on their traditional freehand drawings, but also provides a digital connectivity with downstream processes by vector-type data handling.

This paper is organized as the following order: after related work briefly 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 [13][24] because of their inherent analogy. Many studies tried to use freehand sketch for CAD modeling [8][12][20][21][23][25]. While almost sketch-related research focuses on feature recognition using primitive recognition techniques, 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. It looks a very intuitive tool for designers or animation creators because of its resemblance to physical behaviors. Arvo and Novins [1] proposed so called the *fluid sketch* based on the continuous recognition of freehand sketches and optimal-shape morphing. It also sounds fascinating, but can distort the user's intension because the number of geometric primitives and constraints is limited. We think the fluid sketch will be useful when adopted into engineering-based S/Ws rather than styling-purposed S/Ws. Roth-Koch [22] made overtures of a *B-spline fuzzy array* concept. However, this method has the difficulty of misunderstanding the intention of designers because of its off-line scheme using scanned images.

3 Sketch: Designer's Principal Means of Thinking

There have been great pains to choose two keywords explain the nature of sketch, *guide* and *freehand*, despite of its definitions widely used as follows: "a rough draught of a composition or part of a composition, made in order to satisfy the artist himself on certain point of scale, composition, lighting, etc. [17]" or "a rough drawing or painting in which an artist notes down his preliminary ideas for a work that will eventually be realized with greater precision and detail [4]". Although the meaning of *rough* could remain in above definitions, we wanted to explore more *tangible* words commonly used in the design field as follows:

Definition 1. A design sketch is a guide drawn freehand for a finished work.

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., sketch has been widely used for creative works—"the designer's principal means of thinking [11]", "The act of sketching has long been embraced by architects as a versatile tool for exploratory design [23]". In real product styling, the practical uses of sketch are as follows: 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 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. The major findings on design-sketch elements are as follows:

- Designers do not carry a specific set of geometry primitives. Because they are always dealing with new shapes, it is impossible to work with a limited numbers of primitives.
- A complicated freeform curve is presented with several simple curves. In most cases, the simple unit curve does not have a convexity change.
- In design sketches in 3D perspective views, arbitrarily rotated ellipses are frequently used. An ellipse is generated through one stroke because it is difficult to construct by connecting open curves.
- Straight-lines are rarely used only but in *orthographic* views or as *support* lines because they can cause *visual illusion* –there has been much research on *visual illusion* or *Gestalt Theory* [15][16].
- An exact circle, which is a special case of an ellipse, is limitedly used in mainly orthographic views.

Based on above observations, two propositions related design-sketch elements are made as follows:

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 remarks on design-sketch behaviors throughout our observation:

- Drawing a single curve is composed of several consecutive actions as follows: 1) an *air-sketch*, 2) repetitive touches, 3) a decisional stroke, and 4) a cleaning-up
- An air-sketch is the process accustoms the arm's movement to draw the curve intended (it is quite similar to the golfer's putting in the air).
- Designers settle a final curve with a dark line when its shape is embodied throughout touch repetition with a light line.
- Using *see-through* (grid) box is in common for sketching 3D shapes in a perspective view [6].
- A semi-tracing paper is used when creating new shapes starting from existing sketches.

Like the design-sketch element case, the several propositions about design-sketch behaviors are given as below:

Proposition 3. *Drawing dimensionally accurate curves by controlling a human body, 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 essential for acquiring an intended curve.*

5 Design-Sketch Emulation in Digital Form

In this section, our solution for a digital design-sketch tool is developed. Two major directions are as follows: 1) the interaction between the system and user has to be emulated as a physical sketch as possible, and 2) all the scribbles are treated in the form of mathematical entities not raster images.

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 (we thought that the degrees-of-freedom of a cubic Bezier curve are enough for the designer to fully express curves intended). An ellipses is mapped to a quadratic NURBS curve composed of four rational-quadratic Bezier curves [18]. Under the assumption that users themselves explicitly change the mode of elements during sketching for avoiding vagueness, the two support elements, a straight line and exact circle, are mapped to a linear Bezier curve and a quadratic NURBS curve, respectively.

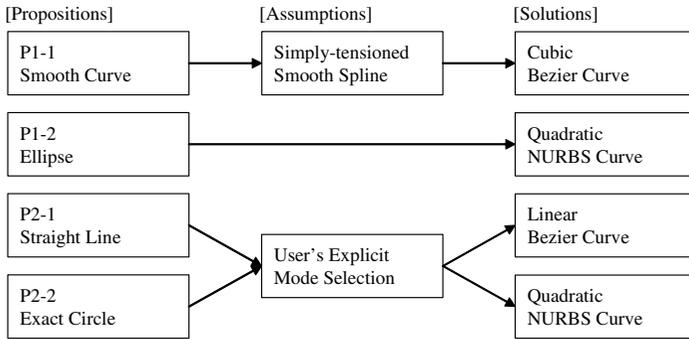


Fig. 1. Design-sketch element mapping

5.2 Design-Sketch Behavior Mapping

In this section, the analyzed designers' sketching behaviors are synthesized into the sketching behavior map. The four propositions related to the designer's behavioral features are mapped to seven solutions based on the vector-type data treatment as shown in Fig. 2.

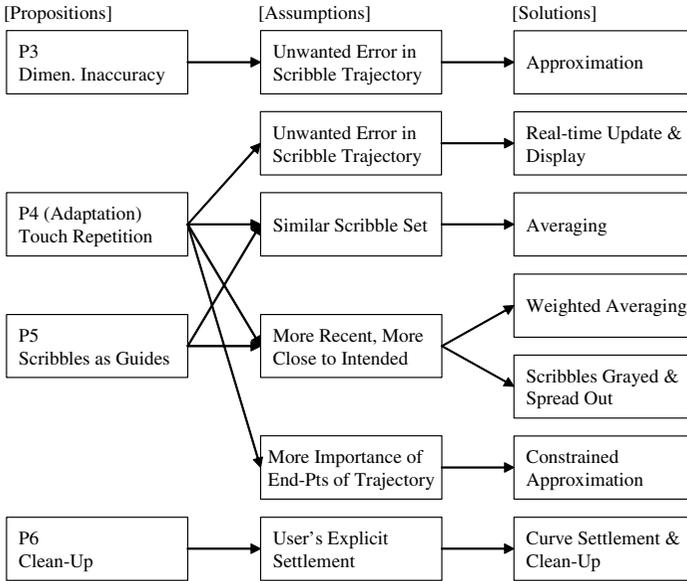


Fig. 2. Design-sketch behavior mapping

Whenever the pen-point changes, the trajectory of a scribble is approximated to the corresponding mathematical model. A trajectory for a smooth curve is fitted to a cubic Bezier curve while the end points are fixed [5][9][18]. The approximation somewhat compensates the dimensional inaccuracy of human drawing by eliminating delicate waves caused by hand shaking—it is a kind of low-frequency filtering or *fairing*. A trajectory for an ellipse is first approximated to general-quadratic curve [7][10], and then converted to a quadratic NURBS curve (only when the general-quadratic curve is discriminated as a *real ellipse* among nine possible types of it [26]). 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. We assumed that more recent scribble is closer to an intended curve than old ones because the designer can adapt his/her drawing while receiving visual feedback. Hence, the settled curve, $\bar{\mathbf{c}}(t)$, is a *weighted average* of a set of scribble curves, $\{\mathbf{c}_i | i = 0, \dots, s - 1\}$, given as:

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

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:

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

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 width of the i^{th} scribble is,

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

where z_{min} is used for the latest scribble, $z_{-\infty}(z[i] \rightarrow a_{-\infty}$ as $i \rightarrow -\infty$), is the upper-bounded width, and $b_z \geq 1$.

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

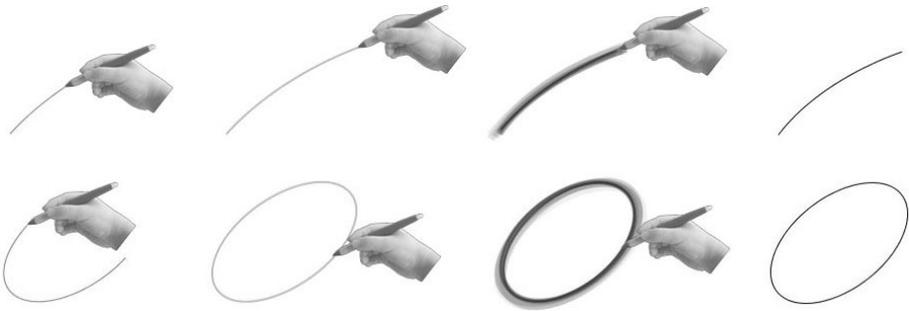


Fig. 3. Sketching simple smooth curve and ellipse

5.3 Implementation

A prototype is implemented as a *Java Applet* program using Java™ 2 Platform Standard Edition (J2SE™), Java 3D™ API 1.3, and JAMA (Java Matrix Package) [14]. The main functionalities are focused on only drawing four design elements: The user can sketch curves using 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™2 9" × 12"), and our Java Applet is available at <http://vr.kaist.ac.kr/~bae>.

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 scribbling guide with existing scribbles grayed and spread out, 3) a natural way to create an ellipse (see Fig. 3), 4) a smooth nature of resultant curves, etc.

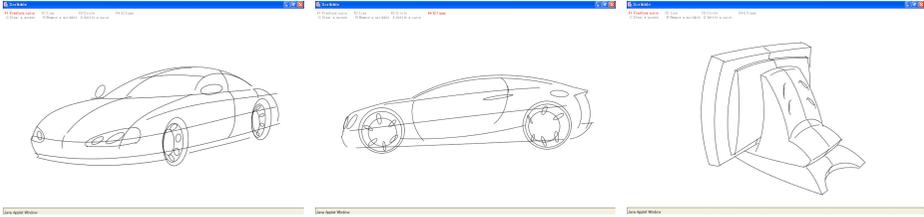


Fig. 4. Selected design-sketches using prototype

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

- Some mentioned about the direction of scribbling. They 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 (in fact, it is one of our next step's research issues).
- A few users commented an unnecessary explicit settlement. For example, a new scribble is quite different from previous ones, the curve settlement can be automatically executed.
- Some suggested not to throw away the scribbles after curve settlement for using them as guides for another shape.
- Many people commented on editing functionalities and GUIs. These kinds of comments are supposed to be useful when developing regular S/Ws later.

7 Discussions

Our strategy of *vectorizing* a scribble trajectory is a kind of *approximation*. We considered if the vectorized curve can be differ from the designer's intention. In spite of this concern, nobody complained of it. Rather, they liked the smooth appearance of resultant curves throughout approximation. It can be thought that the adaptation based on visual feedback is more important than numerical accuracy.

No matter how the designer tries, the adaptation will not be possible if the degrees-of-freedom the model curve has are not enough. 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 among product designers (however, usual graphic designers require more flexible curves for their work). For finding the *optimal* degree, a quantitative study will be followed. Similarly, we are now planning to find optimal parameters for the calculating weighted average, scribbles' gray level and width (in fact, these values can be customized by users as treated as S/Ws' options).

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 (the convexity imposing introduce a NLP problem [19]). We intend to test this strong constraint imposing, but which possibly cause the distortion of the designer's intension (almost designers did not agree with this idea).

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 by including advanced functions such as editing, filleting, concatenating, neutral file in/out, etc., 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. The Proceedings of CGGM (2003)
3. Baudel, T.: A Mark-Based Interaction Paradigm for Free-Hand Drawing. The Proceedings of UIST (1994)
4. <http://www.britanica.com>
5. Choi, B.K.: Surface Modeling for CAD/CAM. Elsevier, NY (1991)
6. Doblin, J.: Perspective: A New System for Designers. Whitney Publications, NY (1956)
7. Eberly, D.H.: 3D Game Engine Design: A Practical Approach to Real-Time Computer Graphics. Morgan Kaufmann, San Francisco (2001)
8. Egli, L., Hsu, C.-Y., Bruderlin, B.D., Elber, G.: Inferring 3D Models from Free-hand Sketches and Constraints. CAD, 29(2) (1997) 101-122
9. Farin, G.: Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide 5th Edition. Academic Press, NY (2002)
10. Faugeras, O.D., Herbert, M.: The Representation, Recognition and Locating of 3-D Objects. Int. J. of Robotics Research 5(3) (1986) 27-52
11. Herbert, D.M.: Architectural Study Drawings. John Wiley & Sons, NY (1993)
12. Hwang, T., Ullman, D.: The Design Capture System: Capturing Back-of-the-Envelope Sketches. J. for Eng. Design 1(4) (1990) 339-353
13. Igarashi, T., Matsuoka, S., Kawachiya, S., Tanaka, H.: Interactive Beautification: A Technique for Rapid Geometric Design. The Proceedings of UIST (1997)
14. <http://math.nist.gov/javanumerics/jama/>
15. Koffka, K.: Principles of Gestalt Psychology. Harcourt Brace, NY (1935)
16. Kohler, W.: Gestalt Psychology: An Introduction to New Concepts in Modern Psychology. Lippincott Williams & Wilkins, NY (1947)
17. Murray, P., Murray, L.: Penguin Dictionary of Art & Artists. Penguin Books, London (1976)
18. Piegl, L., Tiller, W.: The NURBS Book. Springer-Verlag, NY (1995)
19. Pigounakis, K.G., Kaklis, P.D.: Convexity-Preserving Fairing. CAD 28(12) (1996) 981-994
20. Pugh, D.: Designing Solid Objects Using Interactive Sketch Interpretation. The Proceedings of Symposium on Interactive 3D Graphics (1992)
21. 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
22. Roth-Koch, S.: Generating CAD Model from Sketches. The Proceedings of Workshop on Geometric Modelling (2000)

23. Schweikardt, E., Gross, M.D.: Digital clay: Deriving Digital Models from Freehand Sketches. The Proceedings of CHI (2002)
24. Tolba, O., Dorsey, J., McMillan, L.: A Projective Drawing System. The Proceedings of Symposium on Interactive 3D Graphics (2001)
25. Zeleznik, R.C., Herdon, K.P., Hughes, J.F.: SKETCH: An Interface for Sketching 3D Scenes. The Proceedings of SIGGRAPH (1996)
26. Zwillinger, D., Kellogg, B.: CRC Standard Mathematical Tables and Formulae 30th Edition. CRC Press, Florida (1996)