Pegasus: A Drawing System for Rapid Geometric Design

[1]Takeo Igarashi, [2]Sachiko Kawachiya, [1]Hidehiko Tanaka
[1]Dept. of Info. Engneering, [2]Dept. of Info. Science,
Univ. of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan
+81-3-3812-2111 ext 7413
{takeo, tanaka}@mtl.t.u-tokyo.ac.jp,
sachiko@is.s.u-tokyo.ac.jp

Satoshi Matsuoka
Dept. of Math. and Comp. Science,
Tokyo Inst, of Tech.
2-12-1 Ookayama, Meguro-ku.
Tokyo, Japan, +81-3-5734-3876
matsu@is.titech.ac.jp

ABSTRACT
Pegasus is a prototype drawing system for rapid geometric design. Using Pegasus, a user can construct precise geometric diagrams easily without struggling with complicated editing commands. Two novel interaction techniques are introduced: *interactive beautification* and *predictive drawing*. Interactive beautification receives the user's free stroke input and beautifies it by considering possible geometric constraints among segments by generating multiple candidates to prevent recognition errors. Predictive drawing predicts the user's next drawing operation based on the spatial relationship among existing segments on the screen.

Keywords
Geometric Design, Pen-based input, Drawing editor, Predictive interface, Beautification.

INTRODUCTION
Construction of precise geometric diagrams is still a hard, time consuming task. Although commercial CAD and Object-Oriented (OO) drawing systems provide various editing operations, it is quite difficult to find the appropriate combination of editing operations, especially for novice users [2]. For example, the user must combine duplication, flipping, and location adjustment operations to draw a symmetric diagram, and must use duplication and 90 degree rotation to achieve perpendicularity.

To free the users of these complicated editing commands, we propose two interaction techniques, *interactive beautification* [3] and *predictive drawing*, and implement them in a prototype system called Pegasus (Perceptually Enhanced Geometric Assistance Satisfies US!). The combination of these two techniques enables even novice users to draw complicated geometric diagrams (an example is shown in Figure 1) rapidly without using any editing commands.

INTERACTIVE BEAUTIFICATION
Interactive beautification is basically a free stroke beautification with the flollowing added interactive behavior. First, the user draws an approximate shape of his desired segment with a free stroke using a pen or a mouse (Figure 2a). Then, the system infers geometric constraints the input stroke should satisfy by checking the geometric relationship among the input stroke and existing segments (Figure 2b). Finally, the system calculates the coordinates of the beautified segment by solving the simultaneous equations of inferred constraints, and displays the result to the user (Figure 2c).

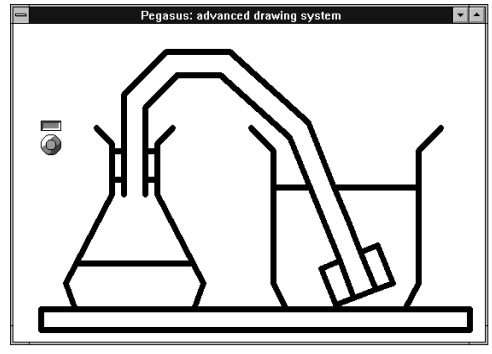

Figure 1: A diagram drawn on Pegasus: this diagram was drawn without any editing commands such as rotation, duplication, or gridding.

Technical problem is that, free stroke input is ambiguous in nature and the beautified segment can frequently fail to correctly capture the user's intention. To solve the problem, interactive beautification infers all possible candidates and allows the user to select one (Figure 2d). The selection is achieved by tapping on the intended candidate segment.

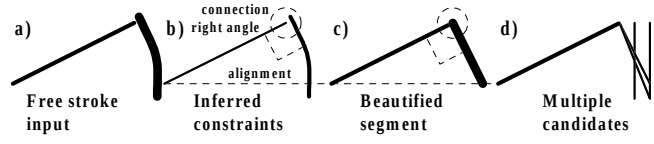

Figure 2: Interactive beautification

PREDICTIVE DRAWING
Geometric illustrations often contain numerous identical local configurations. `Duplicate` command is used in existing drawing editors to generate identical configurations. By contrast, by using interactive

beautification, duplication is *implicitly* achieved by drawing a similar sketch.

Predictive drawing mechanism further assists the construction of identical configurations actively. If the user draws an object whose shape is identical to some existing segment, the system automatically predicts that the user may draw similar segments in the vicinity of the newly drawn segment. The predicted segments are displayed on the screen, and the user can select one by tapping on it if it happens to be the intended segment.

To be specific, the current predictive drawing mechanism works as follows (Figure 3a):

1. When a new segment (trigger segment) is added to the scene as a result of interactive beautification, the system searches for the existing segments (reference segments) whose length and angle are identical to the trigger segment.
2. The system records the spatial relationships among the reference segments and the segments (context segments) directly connected to the reference segments.
3. Finally, the system generates predicted segments in the vicinity of the trigger segment in such a way that the relation between each predicted segment and the trigger segment is identical to that of the context segment and the reference segment.

In addition, the system automatically supports the construction of symmetric diagrams and rotated diagrams by adding flipped and rotated segments to the reference segments (Figure 3b).

When the user selects a predicted candidate by tapping, the system immediately generates the next predictions using the selected candidate as a trigger segment. As a result, he can construct a diagram by successive tapping as long as predicted segments contain the user's intended one. If the user does not find any segments that are useful among the predictions segments, he can seamlessly proceed to the interactive beautification process by drawing the next free stroke.

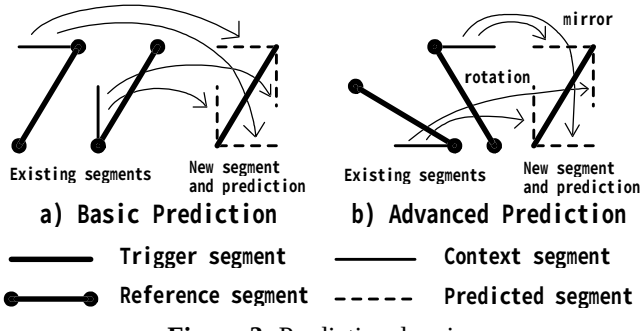

Figure 3: Predictive drawing

RELATED WORK

Existing sketch beautification systems (Apple Newton, SmartSketch, etc.) do contain some features of interactive beautification, but they support only primitive geometric constraints such as connection, while Pegasus supports more complex, global constraints such as symmetry and congruence. Previous beautification systems [8][5] are basically batch-based, which can cause too many errors because of ambiguity in the user's input. Interactive beautification avoids this problem by interactively requesting the user's confirmation for each segment. Automatic inference of geometric constraints can be seen in some research systems [7][4], but they mainly focus on the specification of persistent spatial alignment among movable objects, while we focus on static geometric relationships among line segments in a drawing (parallelism, symmetry, etc.). Prediction mechanisms have been explored in several research efforts [1][6], but their predictions are mainly based on the regularity found in the operation sequence (repeated operation, etc.). Our prediction mechanism is different in that we make use of regularities found in static spatial configurations in a given drawing.

STATUS

The currently implemented constraints are: connection between segments; horizontal or vertical alignment of vertices; congruence, parallelism, and perpendicularity among segments; horizontal and vertical symmetry; and equality of intervals between segments. Curves and arcs are not supported yet but are being planned.

Pegasus is successfully working on small pen computers and Xerox LiveBoards. User experiments showed that the users can draw more rapidly and precisely with Pegasus than with an existing CAD system and an OO drawing editor.

REFERENCES

1. Cypher,A., EAGER: Programming repetitive tasks by example, in *Proceedings of CHI '91*, 33-39, 1991.
2. Igarashi,T., Kawachiya,S., Matsuoka,S., Tanaka,H., In Search for an Ideal Computer-Assisted Drawing System, in *Proceedings of INTERACT '97*, 104-111, 1997.
3. Igarashi,T., Matsuoka,S., Kawachiya,S., Tanaka,H., Interactive Beautification: A Technique for Rapid Geometric Design, in *Proceedings of UIST '97*, 105-114, 1997.
4. Karsenty,S., Landay,J.A., Weikart,C., Inferring graphical constraints with Rockit, in *Proc. of HCI '92*, 137-153, 1992.
5. Kurlander,D., Feiner,S., Interactive Constraint-Based Search and Replace, in *Proceedings of CHI '92*, 606-618, 1992.
6. Masui,T. and Nakayama,K., Repeat and predict - two keys to efficient text editing, in *Proc. of CHI '94*, 118-123, 1994.
7. Maulsby,D.L., Witten,I.H, Kittlitz,K.A., Metamouse: Specifying Graphical Procedures by Example, in *Computer Graphics*, Vol.23, No.3, 127-136, 1989.
8. Pavlidis,T., VanWyk,C.J., An Automatic Beautifier for Drawings and Illustrations, *Computer Graphics*, Vol.19, No.3, 225-234, 1985.