

実用的なスケッチベース図形エディタの実現へ向けて

Towards a Practical Sketch-Based Diagram Editor

中嶋 海介 五十嵐 健夫*

Summary. We present an early prototype system for editing diagrams by a sketching interface. Although diagrams are efficient media to represent ideas, it is hard for novice users to draw them accurately using conventional software. Also, previous research systems aimed at facilitating diagram editing are not yet mature enough for practical use, in that they are far less expressive than conventional software, and force the user unnatural interaction styles to redeem naive algorithms and interfaces. Therefore, our focus is to introduce a new framework for gracefully supporting many types of geometric primitives and constraints, and handling ambiguous user input. Our current implementation is verified to be sufficient for drawing several complex diagrams, and further improvement is in progress.

1 はじめに

「百聞は一見に如かず」という諺にあるとおり、視覚に訴える情報提示は言語による情報提示を往々にして凌駕する。個人がメモを取る際も、文書やプレゼンテーションなどで他人に考えを伝える際も、図は効果的な役目を果たす。しかしながら、紙とペンで容易に描ける図であっても、それをコンピュータ上で綺麗に描くには多大な時間と労力が必要となってしまう。図 1 のような複雑なものの場合、熟練者以外は意図通りに描けないということも起こる。

この問題に対処するため、過去に多くの研究が行われてきており [1, 3, 4, 5], 特にスケッチベースの図形編集インタフェースの有効性が指摘されている。だが、近年ペン入力デバイスが普及し始めているにもかかわらず、実用に耐えるスケッチベースの図形エディタは未だに現れていない。そして、整った図形の編集は依然としてマウスと従来ソフトウェアによって行われているのが現状である。

このような状況を踏まえ、本稿では既存システムの問題点を整理し、その解決方法を提案する。また、それに基づいたプロトタイプシステムを紹介する。

2 既存システムの問題点

人間は古くから紙とペンで情報を記録してきた。コンピュータ上で図形を描く際も、同様のペン型デバイスを使うのが最も自然であると考えられ、既に多くのシステムが提案されている。

Pegasus [3] では、ユーザがストロークを描く度

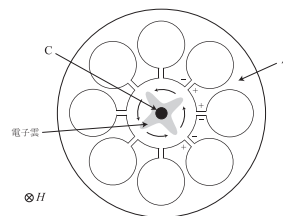


図 1. 従来ソフトウェアや既存のスケッチベース図形エディタで描くのが困難な図の例。複雑な規則性や自由曲線が多く含まれている。

にそれが満たすべき幾何的制約を推論し、整形方法の候補を提示する手法が提案されている。しかし、Pegasus がストロークごとに整形を行うのは認識率の低下を防ぐためであって、人間にとっては意味的なまとまりを描くごとに整形を行う方が自然である。Fluid Sketches [1] では、描画中のストロークをモーフィングで認識結果に近づけることにより、随時ユーザが認識結果を把握・訂正できるようにしている。だが、フィッティングは単純な最小二乗法によって行われており、周囲の図形などは考慮しない。また、これらのシステムは円弧、楕円、自由曲線などに対応していない。

一方、より多種のプリミティブを認識できるものとしては SKIT [5] があるが、図形間の幾何的關係への対応はスナッピング等の単純なものに限られている。現実には、より高度な対称性を考慮した整形が必要となる。

まとめると、既存システムの問題点は以下の二つに大別される。(1) スケッチベースで従来ソフトウェアと同等の表現力を得ることがまだ困難である。具体的には、複雑な規則性のある自由曲線群が描けない。しかし、表現力の向上は認識率の低下とトレードオフの関係にあるため、単純に多種類の幾何プリ

© 2005 日本ソフトウェア科学会 ISS 研究会。

* Kaisuke Nakajima, 東京大学大学院 情報理工学系研究科 コンピュータ科学専攻, Takeo Igarashi, 東京大学大学院 情報理工学系研究科 コンピュータ科学専攻 / 科学技術振興機構 さきがけ

ミティブ・制約をサポートするだけでは済まない。(2) スケッチの曖昧性に対処するためのインタフェースが十分でない。既存システムはユーザの入力を単純なもの(単一ストローク)に限定することで認識ミスの問題を軽減してきたが、それはユーザにとって自然でない。より自然な入力方法を採用すると曖昧性が増すため、ユーザによる訂正の負担も増大するが、そのための対処はまだ十分に研究されていない。

3 解決案

前節で整理した既存システムの問題点を踏まえ、複数ストローク間の幾何制約推論・解消に基づく次のような枠組みを提案する。

3.1 ユーザの入力方法

ユーザはタブレットまたはマウスを用いてキャンバスにストロークを描いていき、意味的なまとまりを描き終えたところでジェスチャにより整形を行う。意味的なまとまりごとの変換は日本語入力システムで既に広く使われており、変換精度に問題がなければストロークごとの整形よりもユーザの負担を軽くできると考えられる。

3.2 整形アルゴリズム

ラフなスケッチを整えるため、既にいくつかのシステム [3, 4] でも使われている制約ベースの整形を行う。

まずユーザの描いた各ストロークを幾何プリミティブにフィッティングする。スケッチには曖昧性があるため、ここでプリミティブを確定することはせず、各プリミティブにどれくらい近いか(最小二乗フィッティングの誤差)を求める。

次に、プリミティブ間に成り立ちうる幾何的制約を推論する。制約としては平行・垂直・合同などの関係をあらかじめ定義しておく。

最後に得られた制約を解き、整形後の座標を求める。プリミティブ・制約の選び方により一般に複数の整形結果が得られるため、各整形結果の美しさ・元ストロークへの忠実さを評価してユーザに提示する。

3.3 誤り訂正インタフェース

推論により制約が多数得られた場合、整形結果の候補も大きく増える。よって、単純にその一覧を提示するとユーザに大きな負担がかかってしまう。そこで、整形結果の候補はスコアの高い方から数個のみ提示して選択させ、その中に望みの結果がない場合は、整形に用いた個々の幾何的制約を表示し、その適用有無を選択させることで細かな指定を可能にする。

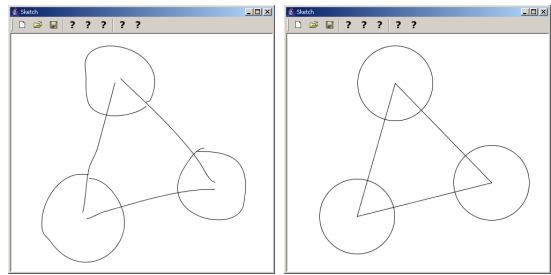


図 2. プロトタイプシステムの使用例。

4 プロトタイプシステム

前節での提案に基づき、Java および制約解消系 Chorus [2] を用いてプロトタイプシステムを実装した。図 2 の使用例のように、ユーザの描いたストローク群をまとめて整形可能である。簡単な実験により、既存のスケッチベースシステムで描くのが困難だった、グローバルな対称性をもつ図形を比較的容易に描けることを確認した。

5 まとめ

実用的なスケッチベース図形エディタの実現へ向けて、先行研究の問題点を整理し、その解決法を提案した。また、それに基づいてプロトタイプを実装し、これまで描くのが難しかった図形が描けることを確認した。今後は未実装の幾何プリミティブ・制約などに対応し、ユーザテストを行う予定である。

参考文献

- [1] J. Arvo and K. Novins. Fluid sketches: continuous recognition and morphing of simple hand-drawn shapes. In *UIST '00: Proceedings of the 13th annual ACM symposium on User interface software and technology*, pp. 73–80, 2000.
- [2] H. Hosobe. A modular geometric constraint solver for user interface applications. In *UIST '01: Proceedings of the 14th annual ACM symposium on User interface software and technology*, pp. 91–100, 2001.
- [3] T. Igarashi, S. Matsuoka, S. Kawachiya, and H. Tanaka. Interactive beautification: a technique for rapid geometric design. In *UIST '97: Proceedings of the 10th annual ACM symposium on User interface software and technology*, pp. 105–114, 1997.
- [4] L. Yan, G. Huang, L. Yin, and W. Liu. A novel constraint-based approach to online graphics recognition. In *Proceedings of the International Workshop on Structural and Syntactic Pattern Recognition*, pp. 104–113, 2004.
- [5] 河合良太, 西川 玲, 佐賀聡人. 手書きスケッチ入力フロントエンドプロセッサ: SKIT. 電子情報通信学会論文誌, J88-D-II(5):897–905, 2005.