インタラクティブなグリッドレイアウト手法の提案

松 田 聖 大[†]　塚 田 浩 二[†,††]

グリッドレイアウトはプロのデザイナの間でも広く用いられる効果的なレイアウト手法だが，紙面に配置する内容に合わせたレイアウトグリッドを作るのは難しい．本研究では，グリッドを用いた作業をインタラクティブに支援する手法を提案し，その実装を行った．本システムでは，ユーザがコンテンツを紙面上で移動・拡大させる際に，コンテンツ全体の配置からレイアウトが持ちうる規則性を解析し，グリッドをインタラクティブに変化させて提示することで，規則性と独自性を兼ね備えたグリッドデザイン手法を実現する．

Interactive Grid-based Layout Method

Shota Matsuda[†] and Koji Tsukada[†,††]

Grid-based layout is an effective way for structuring design materials (contents) and is popular among professional designers. However, even these professionals often have difficulty to design layout grids based on contents. To solve this problem, we propose a novel method that supports users to design layout grid interactively, and developed a prototype system. When a user adds or moves contents on the artboard, the system dynamically analyzes geometric rules of the layout and presents possible layout grids immediately. Thus, the system helps the user to create both structural and unique grid-based designs.

1. はじめに

グリッドレイアウトは，紙面を格子状に分割し，レイアウトに幾何的な規則性を持たせる手法であり，商業誌やグラフィックデザインの分野でプロのデザイナにも幅広く用いられている．グリッドレイアウトは，大量の紙面を画一的に扱うことができるため効率がよく，全体のデザインに統一感を与えることができる．またデザイン手法としては比較てき平易で理解しやすいことから Web デザインでも一般的になってきており，今後も利用機会の拡大が予測される．

グリッドレイアウトを用いるのに最も簡単は方法は，あらかじめレイアウトグリッドを決め，その中に内容を配置していくことである．しかし，汎用なレイアウトグリッドは単調なデザインに帰結しがちであり，また配置するコンテンツの内容と合わないこともある．実際，グラフィックデザインの世界では，デザイナがコンテンツの内容に応じて多種多様なレイアウトグリッドを作り，活用している[4)]．

一方で，紙面に配置する内容に応じてレイアウトグリッドを考えるのは難しい．例えば本文の書体に 12pt の文字サイズ（*size*）と 21pt の行間（*leading*）を用いるとき，レイアウトをグリッド状にするためには，図や空白部分の縦方向の大きさは $leading \times n$ あるいは $leading \times (n-1) + size$ であることが好ましい．また，紙面の大きさと配置するコンテンツの大きさとの関連も考慮しなければならない．こうした制約の中で，規則的で独自性のあるデザインを実現するレイアウトグリッドを考えるのは，本職のデザイナであっても骨の折れる作業である．

そこで，本論文では，コンテンツの状態に応じてインタラクティブに変化するグリッドレイアウト手法を提案する．ユーザがコンテンツを任意に配置・移動させると，全体のレイアウトが持ちうる規則性を解析し，インタラクティブにグリッドを変化させて提示する．ユーザは，このレイアウトグリッドを確認し，必要に応じてレイアウトを変更する．完成したレイアウトグリッドは，本システムからエクスポートすることができ，Illustrator 等の多様な商用ソフトウェアで活用できる（**図 1**）．

† お茶の水女子大学お茶大アカデミック・プロダクション
Ochadai Academic Production,Ochanomizu University
†† 科学技術振興機構 さきがけ
JST PRESTO

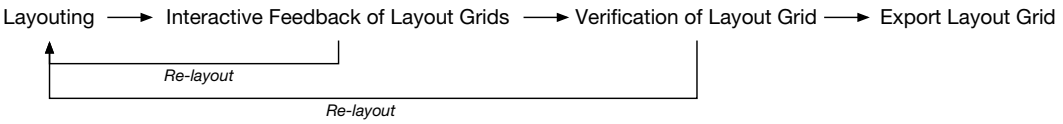

図 1 システムを用いた制作の流れ
Fig. 1 Workflow of the proposed system

2. 関連研究

DTP において，レイアウトグリッドを自動的に生成するシステムは多数提案されている．Feiner らは，紙面上の各種要素（text box や picture）にセマンティックな情報（header や body）を与えることによってレイアウトに制約を設け，レイアウトを制約充足問題として扱う手法を提案している[1)]．また，テンプレートを用いたグリッドレイアウトの制約解決器も提案されている[2),5)]．

一方で，近年，その成り立ちからしてセマンティックな情報を持つ HTML では，手動化のアプローチが採られている．HTML と CSS を用いたグリッドレイアウト・フレームワークである Blueprint☆や The Square Grid☆☆では，セマンティクスを用いたレイアウトグリッドの自動生成ではなく，要素が占めるグリッドの範囲をユーザが手動で指定する．また，Frameless☆☆☆では端末の画面サイズを考慮したリキッドな調整は行われるが，グリッド自体は手動で指定する．

このような自動化と手動化の議論はインタラクションの分野でもしばしば論じられる．Kochhar らはグラフィカルな制作行為（*modeling*）を，創造的な行為（*design*）と明瞭化する行為（*articulation*）から成るとし，インタラクションを 6 つのパラダイムに分類している[3)]．

(1) *Fully manual*：ユーザが全ての決定をし，システムは消極的．
(2) *Constraint-based*：ユーザが全ての決定をするが，選択肢はシステムによって制限される．
(3) *Critic-based*：ユーザが全ての決定をし，システムはデザイン上の欠落を指摘する．
(4) *Improver-based*：デザイン上の欠落をシステムが自動的に指摘し修正する．
(5) *Fully automated*：システムが全ての決定をし，ユーザは消極的．
(6) *CCAD (Cooperative CAD)*：システムは様々な過程でデザインの代替候補を複数生成・提示し，ユーザに決定を促す．

各パラダイム同士に優劣は無いが，レイアウトグリッドの自動生成は，improver-based または fully automated であり，前述した HTML 上のフレームワークは constraint-based に分類できる．我々の提案システムは，レイアウトを自動的に調整するのではなく，ユーザがレイアウトした結果が持ちうる幾何的な規則性をグリッドとしてインタラクティブに提示し，決定をユーザに委ねる点において CCAD である．

3. 提案手法

まず，紙面に配置する要素（コンテンツ）に対して以下の属性を定義する．

- *Margin*：隣り合う要素同士が保つべき最小の間隔．隣り合う要素が異なる margin を持つ場合は大きい方を優先する．
- *Constrained*：要素が配置された位置を固定するかどうか．固定しない場合，生成されるレイアウトグリッドは要素の位置を無視する．
- *Dominant grid*：本文となるべき書体から導き出すレイアウトグリッドの最小単位．本文となる書体は，紙面の大きさとの関係から自動的に判別する．

その上で，本論文が提案する手法は大きく分けて以下の 2 つである．

(1) 周期の提示
(2) レイアウトグリッドの提示

以下ではそれぞれの手法について述べていく．

3.1 周期の提示

各要素のレイアウトがグリッド状になるためには，配置する要素の位置と大きさは，dominant grid に沿っていることが望ましいが，一致に関する明確な基準はない．例えば，要素が dominant grid と整数比になるずれを持つとき，それは一致とは別の規則性を持っていると捉えることもできる．このような，一致以外の規則性を発見するために，要素と dominant grid との関係を「周期」として提示する．

本システムに配置した要素を選択すると，その要素の位置と大きさに合わせて，dominant grid と共に周

☆ http://blueprintcss.org/
☆☆ http://thesquaregrid.com/
☆☆☆ http://framelessgrid.com/

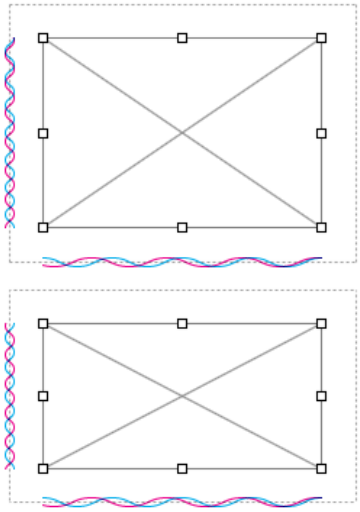

図 2　周期の提示．要素が持ちうる周期をシアン，dominant grid の周期をマゼンタで示している．上：要素の大きさが dominant grid に一致しない場合．下：要素の大きさは dominant grid に一致するが、位置にずれがある場合．

Fig. 2　Showing phases of an element and the dominant grid in cyan and magenta colors. The above figure means the element size does not match the dominant grid. The bellow figure means the element position does not match the dominant grid.

期が提示される．要素の大きさが dominant grid の整数倍であるとき周期が等しくなり，かつ要素の位置が dominant grid に沿うときに完全に重なる．ユーザはこれに合わせて，レイアウトにどのような規則性を持たせるかを判断する（**図 2**）．

3.2　レイアウトグリッドの提示

レイアウトを終えた段階で，システムはそのレイアウトの下で維持される最終的なレイアウトグリッドを生成し，ユーザに提示する．グリッドの生成は詰まるところ制約充足問題であり，解決は困難である．従って，ヒューリスティックな方法を採った．まず，最も大きな寸法と margin を持つ（最も制約の緩い）要素と紙面の寸法から親となるグリッドを生成する．そして，次に大きな要素によって親グリッドを分割し，分割できなければ候補に加え，次に大きな要素に移る．分割できた場合には，次に大きな要素によって親グリッドを分割していく．最終的には複数のグリッドからなる木を得ることになり，最も大きな要素を含み，かつ親グリッドを分割した要素が最も重複する上位 2 つをハイライトしてユーザに提示する（**図 3**）．また，レイアウトグリッドの提示方法の応用として，紙面上の空白部分を提示する機能をシステム上に実装した（**図 4**）．

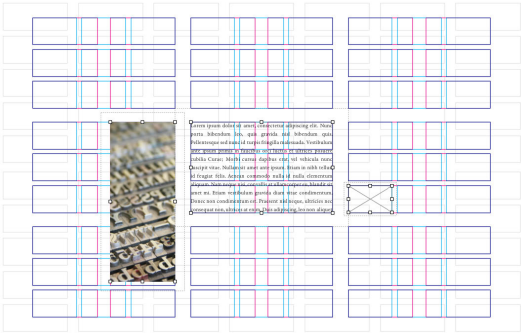

図 3　レイアウトグリッドの提示．この場合，レイアウトグリッドの候補が複数あるため，最も重複する候補をシアン・マゼンタでハイライトしている．

Fig. 3　Presenting layout grids. The system highlights two grid candidates in cyan and magenta.

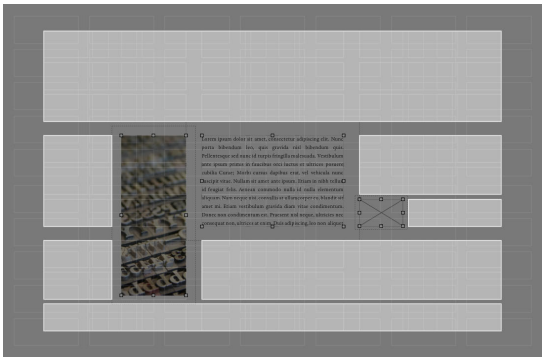

図 4　空白部分の提示．図 3 と同一のコンテンツから生成している．

Fig. 4　Presenting negative spaces for the same contents shown in Fig.3.

4. 実　　装

本システムは，フロントエンドを HTML5 と JavaScript で，低レベルの入出力を Objective-C にて実装しており，Mac OS の Webkit 2 Nightly 上で動作を確認している．**図 5** にシステムのスクリーンショットを示す．

5. 議論と考察

正式な評価実験はまだ行っていないが，本章ではデザイナである筆者や知人らによる試用と主観的な考察を元に，本システムについて議論する．

5.1　グリッドへの吸着

これまで述べたように，本システムは，ユーザの入力に従ってレイアウトグリッドを変化させるのみであり，レイアウト作業は敢えて手動で行う設計としている．この際，要素をレイアウトするときに，提示する規則を満たす位置や大きさへ吸着するが便利であるという意見があった．しかし，グリッドへの吸着を行う

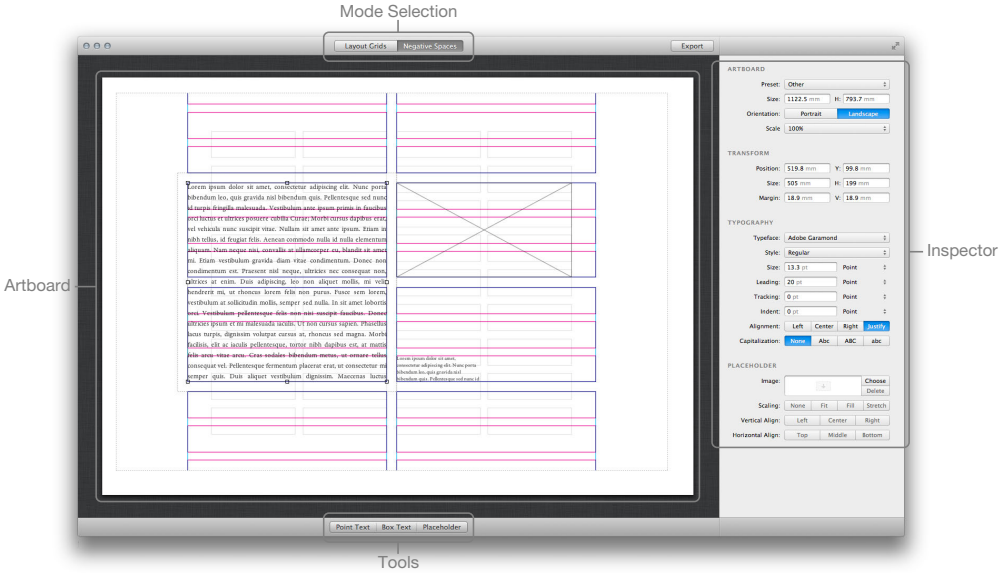

図 5　システムのスクリーンショット
Fig. 5　A screenshot of the proposed system

ということは，吸着する先がレイアウトの解であるという印象を与えてしまうという問題がある．さらに，実装上の問題として，ユーザの入力後に規則性やレイアウトグリッドの計算を一定の時間停止する必要があり，操作性を大きく左右する．現状のシステムでは，妥協案として，Control キーを押しているときのみ計算を停止してグリッドへの吸着を行う機能を持たせた．

5.2　想定ユーザ

ユーザのレイアウト作業に応じて様々な規則性やレイアウトグリッドを提示するということは，既定のものを用いるのに比べて，レイアウトの選択肢が増えるということである．よって，グラフィックデザインの知識が皆無な完全な初心者にとっては，増えた選択肢の中から，望ましいレイアウトを判断するのは難しい可能性がある．今後は，広範なユーザ層に対する試用やユーザテストを通して，本システムの有効性を検証していきたい．

6. ま と め

本研究は，レイアウトの判断をユーザに委ねる一方で，幾何的な規則性を持たせることを目指して，インタラクティブに変化するグリッドレイアウト手法を提案した．しかし，グラフィックデザインに用いされるレイアウトグリッドが持つ役割や規則性は，より多様なものである．今後の課題としては，ユーザにどのような規則性をどこまで提示するかを，ユーザがパラメトリックに選択できるようにすることである．

このように，レイアウトを単に自動化するのではなく，適切な判断材料を積極的にユーザへ提示する手法を追求することによって，新しい形のデザイン支援手法の実現を目指す．

謝辞　本研究の一部は，科学技術振興機構さきがけプログラムの支援を受けた．

参 考 文 献

1) Feiner, S. K.: A Grid-based Approach to Automating Display Layout, *Proceedings on Graphics interface '88*, Toronto, Ontario, Canada, Canadian Information Processing Society, pp.192–197 (1988).
2) Jacobs, C., Li, W., Schrier, E., Bargeron, D. and Salesin, D.: Adaptive Grid-based Document Layout, *ACM SIGGRAPH 2003 Papers*, New York, NY, ACM, pp.838–847 (2003).
3) Kochhar, S., Marks, J. and Friedell, M.: Interaction Paradigms for Human-computer Cooperation in Graphical-object Modeling, *Proceedings of Graphics Interface '91*, Palo Alto, CA, Morgan Kaufmann Publishers, pp.180–191 (1991).
4) Müller-Brockmann, J.: *Grid Systems in Graphics Design*, Arthur Niggli Publishers, Niedersteufen, Switzerland (1981).
5) Schrier, E., Dontcheva, M., Jacobs, C., Wade, G. and Salesin, D.: Adaptive Layout for Dynamically Aggregated Documents, *Proceedings of the 13th International Conference on Intelligent User Interfaces*, IUI '08, New York, NY, ACM, pp.99–108 (2008).