

16 次格子グラフモデルによる複数レイヤー画像の操作

久保田彬仁^{†1} 穴田浩一^{†2} 夜久竹夫^{†1}

複数レイヤーの白黒のラスター画像を 16 次格子グラフでモデル化する。次に複数レイヤーの白黒の 2 次元画像の移動操作アルゴリズムを提案する。このモデル化とアルゴリズムが複数図形の変形操作を行うシステムの開発を容易にする可能性について論じる。さらに、実装の構想を示す。

Operations for Multiple Layer Images with a Hexadecimal Grid Graph Model

AKIHITO KUBOTA^{†1} KOICHI ANADA^{†2} TAKEO YAKU^{†1}

We model multiple layer binary images with hexadecimal grid graphs. Then, we propose algorithms for translations of multiple layer binary images. We discuss that the modeling and algorithms may provide ease of system development for translations of multiple layer images. Furthermore, we show a concept of implementation.

1. はじめに

本研究では白黒のラスター画像を対象とした、表示に適したデータ構造とアルゴリズムについて扱う。ここでは、複数の画像を重ね合わせてひとまとめにしたものを複数レイヤー画像と呼び、複数の画像を効率よく処理するためのデータ構造とアルゴリズムを考える。

通常、2 次元画像の表示や画像に含まれる図形の変形操作、特徴抽出、その他様々な処理を円滑に行うためには効率良く画像の情報を表現することができるデータ構造が必要である。例えばディスプレイなどの端末に画像を表示するために、通常の単一解像度による画像表現を用いていると、一部の解像度を高くしたいとき、画像全体の解像度を高くする必要があるので、大きな“無駄な”計算時間が生じる。

一方、多重解像度に対応したデータ構造を用いて 2 次元画像を表示すると、画像のポリゴン数が減って無駄な計算時間が減る可能性が高まり、なおかつ画像に対する様々な処理に対応しやすくなる。本研究では 2 次元画像を、矩形分割を扱うためのデータ構造で表現することを考える。

先行研究として矩形分割表現のためのいくつかのデータ構造が知られている。不均一な矩形分割を表現可能なデータ構造として Finkel & Bentley は木構造の一種である 4 分木[1]を導入し、Kozminski & Kinnen は矩形双対グラフ[2]の性質を説明した。また、夜久は、8 次格子グラフに基づいた不均一矩形分割のためのデータ構造として octgrid [3]を導入した。octgrid はいくつかの変形に関して双対グラフのデータ構造より計算時間が速いことが知られている。ま

た、Nomaki 等が octgrid の一般化として複数ページのスプレッドシートのような多層表形式のための hexadeci-grid[5]と呼ばれるデータ構造を提案した。hexadeci-grid モデルは octgrid の利点を継承するため、同様に計算時間が速い。octgrid を応用した単一レイヤー画像の操作アルゴリズムはいくつかあるが、複数レイヤー画像の操作アルゴリズムはまだない。

本論文では、複数レイヤー画像の移動アルゴリズムと重ね合わせアルゴリズムを提案する。

2 節では、準備として矩形分割、8 次格子グラフ、16 次格子グラフ、解像度低減化手法について解説を行う。3 節では、移動操作アルゴリズムと図形の重ね合わせアルゴリズムを提案する。さらに、4 節では実装として、システムと実行例を示す。

2. 準備

この節では、3 節以降に必要な予備的概念[3,5]と、3 節以降で使用する既知のアルゴリズム[8,9,10]を解説する。

2.1 矩形分割

(単層型)矩形分割は、例えば図 1 のような平面上のある矩形領域の共通部分のない矩形の集まりによる分割である。矩形分割を構成する矩形をセルと呼ぶ。行もしくは列を表すための周囲に存在する矩形を周辺セルと呼び、それ以外のセルを内部セルという。なお、周辺セルは、 x 方向・ y 方向どちらかの幅を持たないとする。本論文では、矩形分割には周辺セルが存在すると仮定する。また、セルの境界を形成している線を壁という。1 つのセルに対して、上下左右に位置する壁をそれぞれ北壁、南壁、東壁、西壁という。その壁ごとに座標値が付随し、北壁・南壁には y 座標の値が付随し、東壁・西壁には x 座標の値が付随する。図 1 は矩形分割の例である。数値は座標値で、図 1 では、

^{†1} 日本大学
Nihon University
^{†2} 早稲田大学高等学院
Waseda University Senior High School

セル c の北壁には値 0 が付随，南壁には値 1 が付随，西壁には値 1 が付随し，東壁には値 2 が付随している。

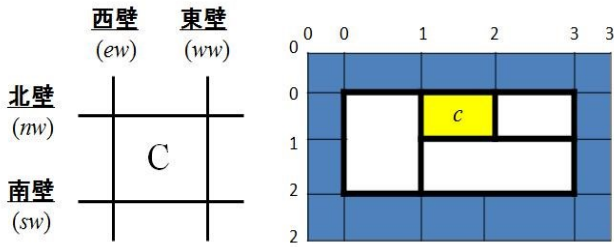


図 1 矩形分割の例

2.2 8 次格子グラフ

文献[3]では，octgrid と呼ばれる 8 次格子グラフに基づいた不均一矩形分割のためのデータ構造を導入した．octgrid 上のいくつかの変形アルゴリズムは，矩形双対グラフのようによく知られているデータ構造の上の対応するアルゴリズムより速いことが知られている。

図 2 は図 1 に対応する octgrid を示している。

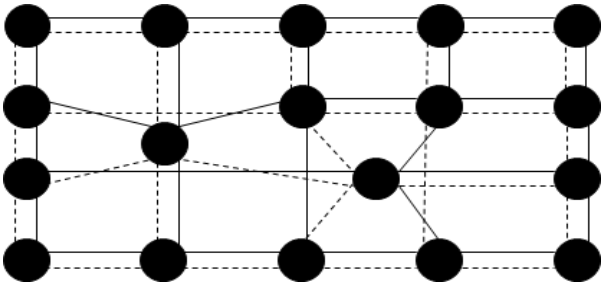


図 2 図 1 に対応する octgrid

2.3 多層型矩形分割と 16 次格子グラフ

文献[5]では，例えば図 2(a) のような多層型矩形分割のための立体型 16 次グラフ（超格子グラフ）に基づいた hexadeci-grid (図 3(b)) と呼ばれるデータ構造を提案した．なお，多層型矩形分割にも各層の周囲に周辺セルがあり，一番上の層と一番下の層を周辺セルに相当する周辺層があると仮定する。

hexadeci-grid は octgrid を複数レイヤーに一般化したもので，罫線を維持する変形アルゴリズムに適している．図 2 で例を示す。

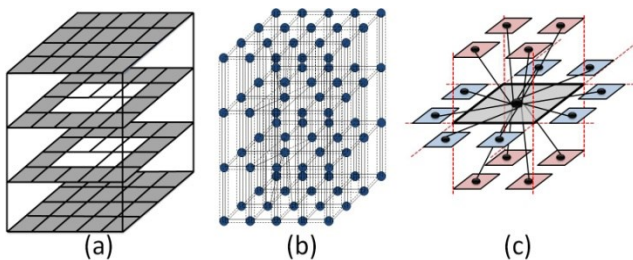


図 3 (a) 対象とする多層型矩形分割. (b) (a) に対応する hexadeci-grid. (c) 内部ノードの周囲のリンク.

hexadeci-grid は以下のように定義される[5].

以下で hexadeci-grid を形式的に定義する。

定義 2.3.1

D を多層矩形分割とする。

D に対する hexadeci-grid は多重無向グラフ $G_D=(V_D, L, E_D, A_D, \alpha_D)$ である。

ここで， $V_D, L, E_D, A_D, \alpha_D$ は以下のとおりである。

V_D : 多層の矩形分割 D に対応するノードの集合。

$L = \{\text{north wall edge, south wall edge, east wall edge, west wall edge, northeastern corner edge, northwestern corner edge, southeastern corner edge, southwestern corner edge}\}$.

E_D : 規則[5]に基づいたエッジの集合。

A_D, α_D : V_D のすべてのノードの属性の集合と属性関数。

2.4 セルの合併アルゴリズム

octgrid で表されている単層型矩形分割内の 2 つ並んだセルを合併する $O(1)$ -アルゴリズムとして “UnifyCell” [4] が提案されている。

図 4 は “UnifyCell” の例である。

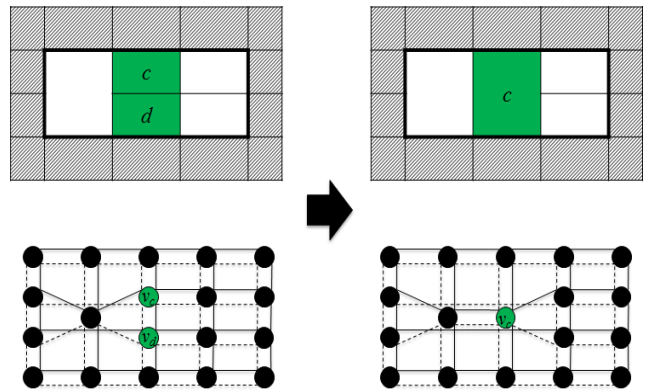


図 4 “UnifyCell” によるセルの合併の例

2.5 解像度低減化

“unifycell” を用いることで，矩形分割内のセル数を減少させ，セルに対応しているピクセルの数も減少させることができる．文献[8,9]で shindo 等は，3D 地形図の解像度低減化アルゴリズム[6]を応用し，2 次元画像に対する以下のような 2 つの解像度低減化アルゴリズムを提案した。

Algorithm “Reduction8” (ヒルベルト走査法を用いた合併→垂直方向合併→水平方向合併)

入力

G_D : 原画像の矩形分割 D に対応する octgrid ($2^n \times 2^n$ サイズ)

出力

G_E : 解像度が低減化された矩形分割 E に対応する octgrid

方法

1. 初期化 $G_E \leftarrow G_D$.
2. G_E においてヒルベルト曲線に沿って可能な限り “UnifyCell” によりセルを合併する。

3. G_E において垂直方向に沿って可能な限り“UnifyCell”によりセルを合併する.
4. G_E において水平方向に沿って可能な限り“UnifyCell”によりセルを合併する.

2.6 移動アルゴリズム

文献[10]で kubota 等は, 2次元画像に対する解像度低減化アルゴリズム[8,9]を用いた, 以下のような単層型矩形分割の図形操作アルゴリズムを提案した.

Algorithm “Translation8”

入力 G_D : 不均一型矩形分割 D に対応する octgrid($2^n \times 2^n$ サイズ).

x : 水平方向に移動する距離.

y : 垂直方向に移動する距離.

出力 G_E : 図形が移動した不均一型矩形分割 E に対応する octgrid.

方法

Step1 初期化 $G_E \leftarrow G_D$.

Step2 G_E を均一な矩形に変換する.

Step3 G_E のオブジェクトを構成するセルを, 水平方向に x , 垂直方向に y 移動する.

“Reduction8”を用いて G_E を不均一な矩形に変換する.

3. 操作アルゴリズム

本節では, 初めに 2.4 に示した単層型矩形分割上の図形の移動操作アルゴリズム“Translation8”[10]を応用し, hexadeci-grid で表された複数レイヤー画像の移動アルゴリズム“Translation16”を示す. 次に 2.4 で示した単層型矩形分割上の図形の解像度低減化アルゴリズム“Reduction8”を応用し, 重ね合わせアルゴリズム“Projection16”を示す. さらに, 2 つを合わせて移動と重ね合わせアルゴリズム“TranslationAndProjection16”を示す.

Algorithm “Translation16”

入力

G_D : 多層型矩形分割 D に対応する

hexadeci-grid (原画像 1 枚は $2^n \times 2^n$ サイズ)

d_x : x 方向に移動する距離

d_y : y 方向に移動する距離

z : 対象とする層番号

出力

G_E : z 番目の層で, 図形が x 方向に d_x , y 方向に d_y 移動した多層型矩形分割 E に対応する hexadeci-grid

方法

Step 1: 初期化 $G_E \leftarrow G_D$

Step 2: 単層型矩形分割上の図形の移動操作アルゴリズム[10]と同様に, 層 z の画像を x 方向に d_x , y 方向に d_y 移動させる.

Step 3: 層 z の上下のリンクを付け替える.

以下の図 5 で“Translation16”の実行に伴う z 番目の層の変化を示す.

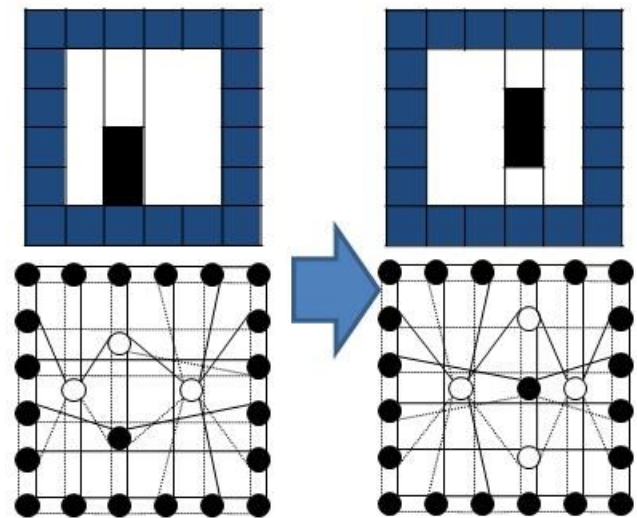


図 5 “Translation16”を用いた移動の例

次に, 重ね合わせアルゴリズムを示す.

Algorithm “Projection16”

入力

G_D : 原画像の多層型矩形分割 D に対応する hexadeci-grid ($2^n \times 2^n$ サイズ), k 層とする(内部は $k-2$ 層).

出力

G_E : D の 2 層目から $k-1$ 層目までの画像を重ねあわせた画像を新たに k 層に配置し, $k+1$ 層を周辺層とする $k+1$ 層の多層型矩形分割 E に対応する hexadeci-grid

方法

Step 1: 初期化 $G_E \leftarrow G_D$

Step 2: 上下方向のリンクを用いて, 2 層目から $k-1$ 層目までの画像を重ね合わせた画像を作って, その画像を k 層に配置する.

Step 3: “Reduction8”と同様の方法で k 層目の画像を不均一な矩形分割に変形する.

Step 4: 上下のリンクを付け替える.

以下の図 6 で“Projection16”の入力層(左上, 右上)と途中経過(左下)と出力層(右下)に対応する矩形分割を示す.

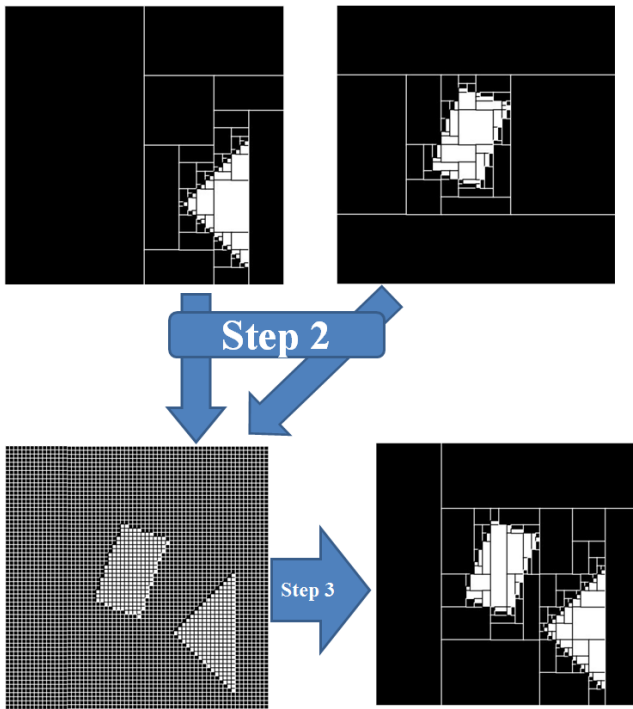


図6 “Projection16”の実行概念の例(周辺セルは省略)

さらに, ”Translation16”と”Projection16”を合わせたアルゴリズムを示す.

Algorithm “TranslationAndProjection16”

入力

- G_D : 原画像の多層型矩形分割 D に対応する hexadeci-grid ($2^n \times 2^n$ サイズ) k 層とする(内部は $k-2$ 層).
- d_x : x 方向に移動する距離
- d_y : y 方向に移動する距離
- z : 対象とする層番号

出力:

G_E : D の 2 層目から $k-1$ 層目までの画像を重ねあわせた画像を新たに k 層に挿入し, $k+1$ 層を周辺層とする $k+1$ 層の多層型矩形分割 E に対応する hexadeci-grid

方法

- Step 1: 初期化 $G_E \leftarrow G_D$
- Step 2: G_E において, ”Translation16”を用いて図形を x 方向に d_x , y 方向に d_y 移動させる.
- Step 3: ”Projection16”を用いて, 2 層目から $k-1$ 層目までの画像を重ね合わせた画像を作って, その画像を k 層に配置する. 複数レイヤーの画像を重ね合わせる.
- Step 4: ”Reduction8”と同様の方法で k 層目の画像を不均一な矩形分割に変形する.
- Step 5: リンクを付け替える.

以下の図 7 で”TranslationAndProjection16”の入力層(左上, 右上)と各層の移動結果(左中, 右中)と途中経過(左下)と出力層(右下)に対応する矩形分割を示す.

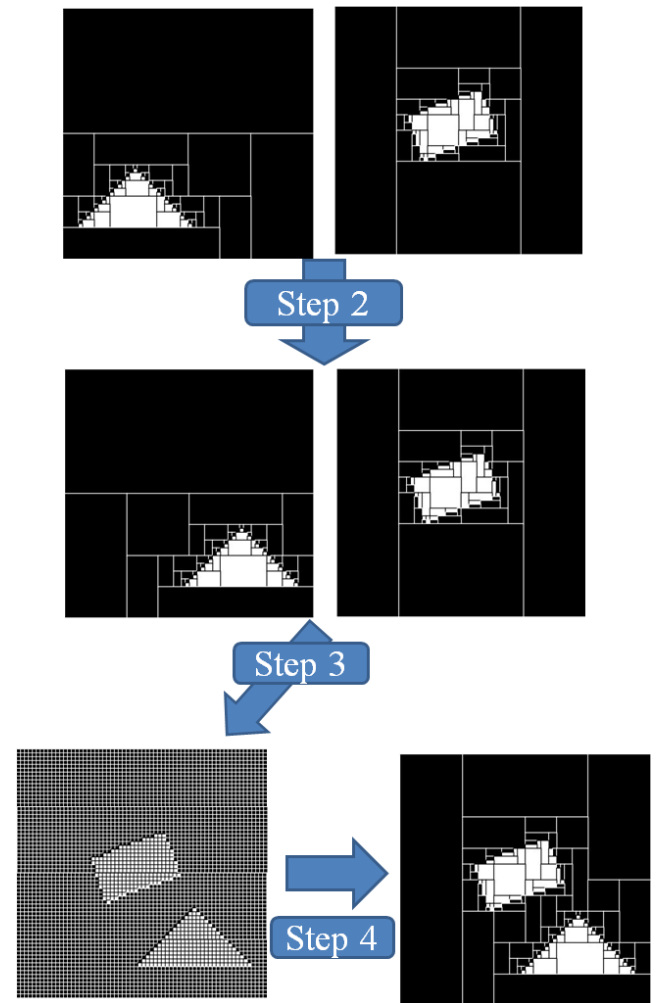


図7 “TranslationAndProjection16”の実行概念の例(周辺セルは省略)

複数の図形が含まれる画像を考える. その画像の中で, 個々の図形を独立に移動させる場合に, 移動させたい個々の図形 (たとえば 2 個. (図 7)) を別々の層 (この場合は 2 層) に配置してアルゴリズム ”Translation16” や ”TranslationAndProjection16” を適用することが可能である. この方法により, 図形の操作システムの開発が容易になると思われる.

4. 応用

本節では, 3 節のアルゴリズムを応用した, システムについて述べる. はじめに, hexadeci-grid のデータフォーマットの例を解説する. 次に, システムのデータ流れ図を示し, さらに開発中のシステムの実行画面を示す.

- 2) Kozminsky, K. and Kinnen, E.: Rectangular Duals of Planar Graphs, *Networks* 15, pp. 145-157, 1985.
- 3) Yaku, T. "Representation of Heterogeneous Tessellation Structures by Graphs," *Memoir of WAAP Meetings* 108, 6p, Dec. 2001. In <http://www.waap.gr.jp/waap-rr/waap-rr-01-013.pdf>
- 4) Kirishima, T. Motohashi, T. Tsuchida, K. and Yaku, T.: Table Processing Based on Attribute Graphs, *Proc. 6th IASTED Conf. SEA*, pp. 317-320, 2004.
- 5) Nomaki, K. Arita, T. Koka, S. Tsuchida, K. and Yaku, T.: A Hexadecimal Grid Graph Model for the Multiply Layered Tabular Forms, *Proc. 2010 Internat. Conf. Computer & Software Modeling*, pp.40-44, 2010.
- 6) Akagi, G. Anada, K. Koka, S. Nakayama, Y. Nomaki, K. and Yaku, T.: A Resolution Reduction Method for Multi-resolution Terrain Maps, *ACM SIGGRAPH 2012 Posters*, 2012.
- 7) Anzai, K. Koka, S. and Yaku, T.: H4CODE 1.2 Reference Manual, Working Group of Automata and Its Applications Research Report, 12-002, 4p, May, 2012. URL: <http://www.waap.gr.jp/waap-rr/waap-rr-12-002/index.html>
- 8) 神藤悠希, 穴田浩一, 夜久竹夫: 8次格子グラフによる2次元画像の解像度低減化, 情報処理学会研究報告 *IPSJ SIG Technical Report*, Vol. 2012-MPS-91, No.29, 2012.
- 9) Shindo, Y. Anada, K. Kikuchi, T. Koka, S. and Yaku, T.: Reduction of Resolution for Binary Images by an Octal Grid Graph Representation Model, *Proc. 12th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2013)*, pp.417-422, 2013.
- 10) Anada, K. Koka, S. Kubota, A. Shindo, Y. and Yaku, T.: The Number of Cells in Regions Shifted on 2D Images Represented by Raster Data with Heterogeneous Parts, *Proc. 14th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD 2013)*, pp490-495, 2013.
- 11) <http://www.yakulab.org/archives/2013-12-11-MPS96>