

## 平面図形の8次格子表現 Octgrid Representation of 2D Objects

神藤 悠希<sup>†</sup>  
Yuki Shindo

穴田 浩一<sup>‡</sup>  
Koichi Anada

夜久 竹夫<sup>†</sup>  
Takeo Yaku

### 1. はじめに

本研究では平面図形の表示に適したデータ構造を扱う。

平面図形を均一矩形分割で表現する場合、一部の解像度を高くするために、全体の解像度を高くする必要がある。そのため、ディスプレイ等に表示するために大きな計算時間を要し、作業効率の低下などが考えられる。

先行研究では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。Finkel & Bentley は木構造の一種である4分木[2]を、Kozminski & Kinnen は矩形双対グラフ[3]を導入した。また、夜久等は、octgrid [5]と呼ばれる不均一矩形分割のための8次格子グラフに基づいたデータ構造を導入した。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。

本論文は、octgrid の技法を用いて平面図形を不均一な矩形分割で表現する手法を示す。

3節では、平面図形を不均一な矩形分割で表現するための方法を示す。4節では、例を用いて我々の方法により表現された平面図形の解像度と、既存の方法であるヒルベルト走査法により表現された平面図形の解像度の比較を行う。

### 2. 準備

#### 2.1 octgrid [5]

octgrid [5]は8次格子グラフの一種で、不均一矩形分割のためのデータ構造の1つである。octgrid の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な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。

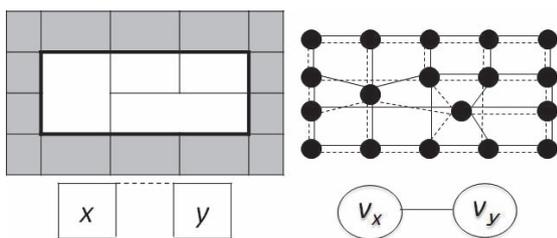


図1 不均一矩形分割(左)と対応する octgrid(右)

#### 2.2 ヒルベルト走査法

Hilbert はフラクタル図形の一つで、空間を覆い尽くす空間充填曲線の一つである“ヒルベルト曲線” [1]を提示した。図2に示されるように、この曲線は2次元平面上的のタイルを隙間なく通過する。

さらに、このヒルベルト曲線は2次元空間内のデータを走査するためによく用いられている[4]。その走査方法は、“ヒルベルト走査法”と呼ばれ、画像処理分野において画

像圧縮などに広く適用されている。

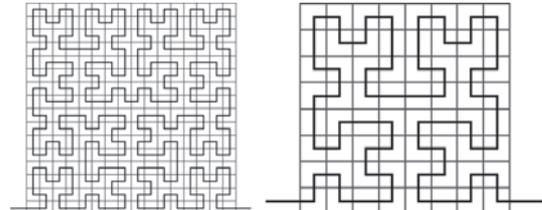


図2 ヒルベルト曲線の例

### 3. 平面図形の表現

本節では、octgrid のセルを平面図形のピクセルに対応させ、図形の特徴を保ったままセルの総数を減らすための以下の2つのアルゴリズムを提案する。

- ① ヒルベルト曲線、垂直および水平方向の順に走査しセルを合併していく。
- ② ヒルベルト曲線、水平および垂直方向の順に走査しセルを合併していく。

また、この方法は、様々な曲線に沿って走査が行えるという octgrid の利点を活かしている。

#### 3.1 セルの合併[6]

矩形分割のセルの合併操作を行うアルゴリズムである“UnifyCell”[6]を用い、octgrid に対応している矩形分割内の“2つ並んだ”セルを合併する。このアルゴリズムを用いることによって、矩形分割内のセルの数を減少させ、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。図3は“UnifyCell”の例である。

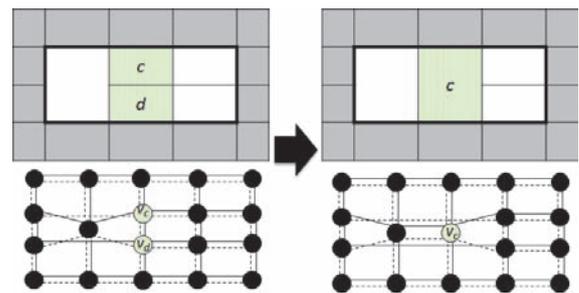


図3 “UnifyCell”によるセルの合併の例

#### 3.2 解像度低減化

我々は3D地形図の解像度低減化アルゴリズム[7]を応用し、平面図形の解像度低減を行う。

**Algorithm 1** (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  $G_E \leftarrow G_D$ .

<sup>†</sup> 日本大学 Nihon University

<sup>‡</sup> 早稲田大学高等学院 Waseda University Senior High School

- 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。

#### Algorithm 2 (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:  $G_E$ において水平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. 比較

例を用いて、図形の特徴を保ったまま、ヒルベルト走査法を用いたセルの合併による解像度低減を行った平面図形のセル数と、我々の Algorithm 1, 2 を用いた場合のセル数の比較を行う。

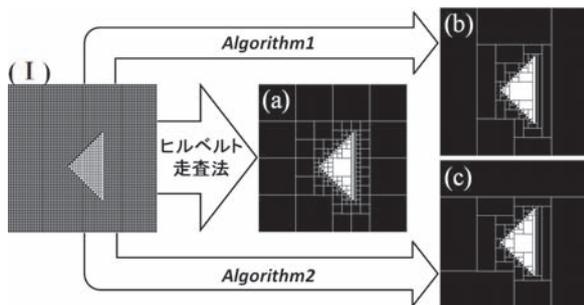


図 4 (I)に対しヒルベルト走査法を用いたセルの合併と Algorithm 1, 2 を適用し解像度低減を行った例。

図 4 の (a) は三角形 (I) に対しヒルベルト走査法を用いたセルの合併の結果で、セルの数は 220 である。次に (I) に対し Algorithm 1, 2 を適用し (b), (c) を生成する。(b) は垂直方向、水平方向の順で合併を行った結果で、セルの数は 78 である。(c) は水平方向、垂直方向の順で合併を行った結果で、セルの数は 81 である。なお (a) に対しての (b), (c) の割合は 35.45% と 36.82% である。

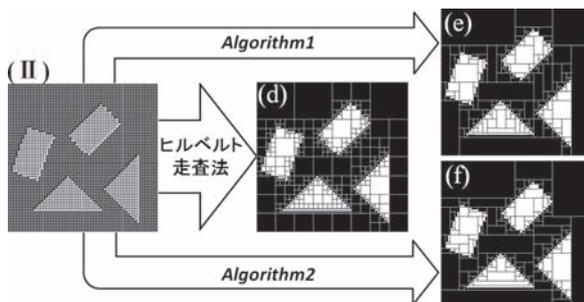


図 5 (II)に対しヒルベルト走査法を用いたセルの合併と Algorithm 1, 2 を適用し解像度低減を行った例。

図 5 の (d) は 4 つの図形 (II) に対しヒルベルト走査法を用いたセルの合併の結果で、セルの数は 577 である。次に (II) に対し Algorithm 1, 2 を適用し (e), (f) を生成する。(e) は垂直方向、水平方向の順で合併を行った結果で、セルの数は 286 である。(f) は水平方向、垂直方向の順で合併を行った結果で、セルの数は 279 である。なお (d) に対しての (e), (f) の割合は 49.57% と 48.35% である。

表 1 は以上をまとめたものである。

X: ヒルベルト走査法		Y: Algorithm 1, 2		Y/X
(a)	220 セル	(b) (Algorithm 1)	78 セル	35.45%
		(c) (Algorithm 2)	81 セル	36.82%
(d)	577 セル	(e) (Algorithm 1)	286 セル	49.57%
		(f) (Algorithm 2)	279 セル	48.35%

表 1 Algorithm 1, 2 と、ヒルベルト走査法で低減された平面図形のセル数の比較。

以上より、我々の方法はヒルベルト走査法を用いたセルの合併よりも大幅なセル数の減少を可能にする。

#### 5. まとめ

平面図形を表現するための octgrid と呼ばれる 8 次格子グラフを紹介した。さらに、図形の特徴を保ったまま効果的に解像度を減らすためのアルゴリズムを提案した。

また、例を用いて我々の方法と、既存の方法であるヒルベルト走査法で低減された平面図形の解像度との比較を示し、結果として我々の方法が 50% 以上ものセル数の減少を可能にすることを示した。

今後、この octgrid の技法を応用し様々な平面図形の変形を行う方法を研究し、実装する。

#### 謝辞

本研究にあたり貴重な助言を頂いた東海大学の杉田公生先生、東洋大学の土田賢省先生、日本大学の野牧賢志氏に深く感謝いたします。

#### 参考文献

- [1] D. Hilbert, “Über die stetige Abbildung einer Linie auf ein Flächenstück,” *Math. Ann.* 38, pp. 459–460 (1891).
- [2] R.A. Finkel and J.L. Bentley, “Quad Trees: A Data Structure for Retrieval on Composite Keys,” *Acta Informatica* 4 (1) pp. 1–9 (1974).
- [3] K. Kozminsky, and E. Kinnen, “Rectangular Duals of Planar Graphs” *Networks* 15, pp. 145–157 (1985).
- [4] S. Kamata and Y. Hayashi, “Region-based scanning for image compression,” *IEEE International Conference on Image Processing*, pp. 895–898 (2000).
- [5] T. Yaku, “Representation of Heterogeneous Tessellation Structures by Graphs,” *Memoir of WAAP Meetings* 108, 6p, Dec. 2001. In [http://www.waap.gr.jp/waap-memoir/waap108/waap108\\_02-yaku/011201waap108table-rep-doc.pdf](http://www.waap.gr.jp/waap-memoir/waap108/waap108_02-yaku/011201waap108table-rep-doc.pdf) (2001).
- [6] T. Kirishima, T. Motohashi, K. Tsuchida, and T. Yaku, “Table Processing Based on Attribute Graphs,” *Proc. 6th IASTED Conf. SEA*, pp. 317–320 (2004).
- [7] G. Akagi, K. Anada, S. Koka, Y. Nakayama, K. Nomaki, and T. Yaku, “A Resolution Reduction Method for Multi-resolution Terrain Maps,” *SIGGRAPH 2012 Posters*, to appear.