

# 地形形状チップス： インタラクティブな2次元ゲームフィールドビュー操作系

寺井真紀<sup>†</sup> 藤木 淳<sup>††</sup>  
鶴野玲治<sup>†††</sup> 富松 潔<sup>†††</sup>

## Topography Shape Chips : Interactive view angle interface for 2D Game Field

MAKI TERAI,<sup>†</sup> JUN FUJIKI,<sup>††</sup> REIJI TSURUNO<sup>†††</sup>  
and KIYOSHI TOMIMATSU<sup>†††</sup>

### 1. デザインコンセプト

平面的なコンピュータゲームフィールドの表示として、図1に示すような上面図に高さを加えたような表示法（以下、見下ろし型視点表示）や、図2のように真横から見た表示法（以下、サイドビュー表示）がある。両者はそれぞれ異なる利点を持つため、同時に使うことで新しい表現を提供することができると考えられる。

本研究ではインタラクティブに両者を行き来し、2次元の操作系で3次元のフィールド形状を取り扱う方法を試行する。錯視・錯覚を利用した3次元作図法としては Incompatible BLOCK<sup>1)</sup> があるが、ここでは、見下ろし型視点表示を持つゲームでの利用を前提

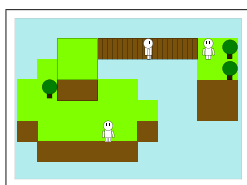


図1 見下ろし型視点表示

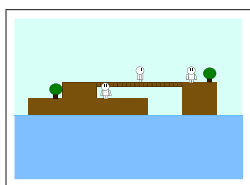


図2 サイドビュー表示

<sup>†</sup> 九州大学芸術工学部芸術情報設計学科  
School of Design, Kyushu University

<sup>††</sup> 九州大学大学院芸術工学府  
Graduate School of Design, Kyushu University

<sup>†††</sup> 九州大学大学院芸術工学研究院  
Faculty of Design, Kyushu University

とするため、ユーザは見下ろし型視点表示でフィールドをデザインできることが好ましい。その表示において色の違いや輪郭線の有無を考慮してタイルを並べて立体を認識できるフィールドをデザインできるようにする。さらに、3次元構造を視点に依存して変換することで、両者の利点を兼ね備えた操作系の実現を目指している。

### 2. インタフェース

「地形形状チップス」では、ユーザは見下ろし型視点表示において、タイルを配置することで地形形状を作り、立体形状を得る。図3は「地形形状チップス」の画面例である。ユーザは画面右に並ぶアイコンからタイルを選択し、キャンパス内でクリックしてタイルを配置することで、見下ろし型視点表示での地形形状をデザインする。また、右ドラッグで空間を回転し、インタラクティブにサイドビュー表示へ移行することができる。図4は図3のタイル配置で立体化したものである。

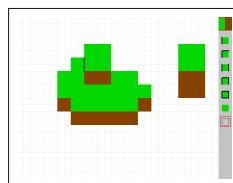


図3 画面例

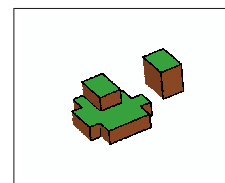


図4 移行する様子

### 3. アルゴリズム

本システムでは空間の回転開始時に見下ろし型視点表示の2次元の地形形状を立体化する。次に、立体化された地形形状を視点方向に応じて変形して描画する仕組みを取る。以下に、それらのアルゴリズムを述べる。

#### 3.1 地形形状の立体化

地形形状の立体化では、2次元平面上のタイル配置のデータから3次元のボクセルデータを定義する。まずワークスペースの全てのタイルを走査し、緑タイルが形成する閉領域を求める。閉領域をひとつのグループとし、緑タイルを閉領域グループに属させる。次に全ての閉領域に対して、それぞれ以下の処理を行う。

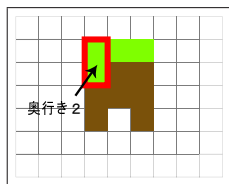


図 5 列ごとの奥行き算出

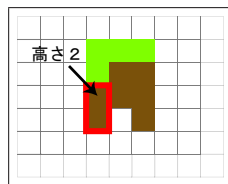


図 6 列ごとの高さ算出

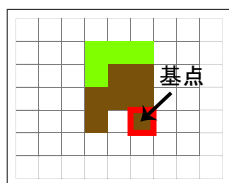


図 7 最高値算出

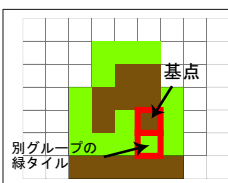


図 8 高さ修正

#### (1) 列ごとの奥行き算出 (図 5)

列ごとにタイルを走査する。連続している緑タイルの個数を求め、この個数を奥行き値とする。

#### (2) 列ごとの高さ算出 (図 6)

列ごとにタイルを走査する。緑タイルの下に隣接する茶タイルの個数を求め、個数を列の高さとする。

#### (3) 最高値算出 (図 7)

全ての列の高さが求まったら、閉領域が持つ列の中で最も高い高さをその閉領域の高さとする。また、最高の高さをもつ列の、茶タイルの一番下のタイルを基点タイルとする。この基点タイルの2次元平面上の座標を基準位置とし、各列

の高さと奥行きデータより3次元データを定義する。

#### (4) 高さ修正 (図 8)

基点タイルの下に緑タイルが隣接し、かつその緑タイルが別の閉領域グループであるときは、階段状の形状を作る。そのため、茶タイルが属する閉領域の最高の高さに、緑タイルが属する閉領域の最高の高さを加える。

#### 3.2 地形形状の変形

本システムでは視点に依存して形状を変形させる。見下ろし型視点表示では立方体の上面と側面の大きさが等しくなるように表示する。立体化した時に、このような表示を実現するため、本システムでは見下ろし型視点表示を、図 9 から図 10 のように立方体の上面と側面の大きさが等しくなるように変形させる。また、サイドビュー表示へシームレスに移行するため、視点に応じて変形率を変更する。

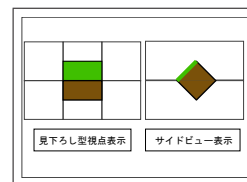


図 9 変形前

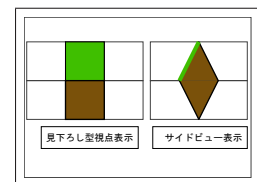


図 10 変形後

### 4. 結果・考察

本研究では、見下ろし型視点表示でデザインしたフィールドをインタラクティブにサイドビュー表示に行き来できるシステムを開発した。本技法はサイドビュー表示移行時にフィールドから立体形状を得るが、ユーザの意図しない形状となる場合もある。しかし、そのことを好ましく捉えるユーザもいた。また、立体形状の変形は異なる視点表示に移行する際に行われるが、その事実を認識したユーザはほとんどおらず、本技法が異なる2つの視点表示間をシームレスに移行可能となったことを確認できた。今後は本技法をフィールドだけでなく、キャラクタやオブジェクトなどに適用することでよりゲームに応用できると考えている。

#### 参考文献

- 1) Fujiki, J., Ushiyama, T., Tomimatsu, K., " In-compatible BLOCK: Wonders Accompanied Interface ", ACM SIGCHI2006, April, 2006