

## Design research of subjective–interaction for 3DCG tools

藤木, 淳

<https://doi.org/10.15017/459602>

---

出版情報 : 九州大学, 2006, 博士 (芸術工学), 課程博士  
バージョン :  
権利関係 :

## 第3章

Incompatible BLOCK : 3次元CGモデリングのための  
主観指向インタラクションデザイン2

### 3.1 はじめに

本章では、引き続き3次元CGモデリングツールのための主観指向インタラクションデザインについて述べる。前章の考察を踏まえ、立体の配置と着色における主観指向インタラクションデザインの改良を行う。

「Incompatible BLOCK」は、「はいはいいんき」と同様、立方体の組み合わせで形状の生成を行い、形状や床、背景に対して2次元の操作で線を描くことができる3次元CGモデリングツールである。だまし絵のような立方体の配置とペンのサイズ変更操作はよりユーザを楽しみやすい操作へと導くデザインである。

図3.1に「Incompatible BLOCK」の画面を示す。「Incompatible BLOCK」ではキャンバス内にアイコンが置かれる。画面右には並ぶペンツールが形状や床や背景に線を描くための描画ツールで8色から選択する。「Incompatible BLOCK」は後述する「影」の黒色との混同を避けるため黒色が外されている。画面左には立方体を操作するための手ツール、空間を回転させるための回転ツール、空間をズームするためのズームツール、セッションを終了するドア

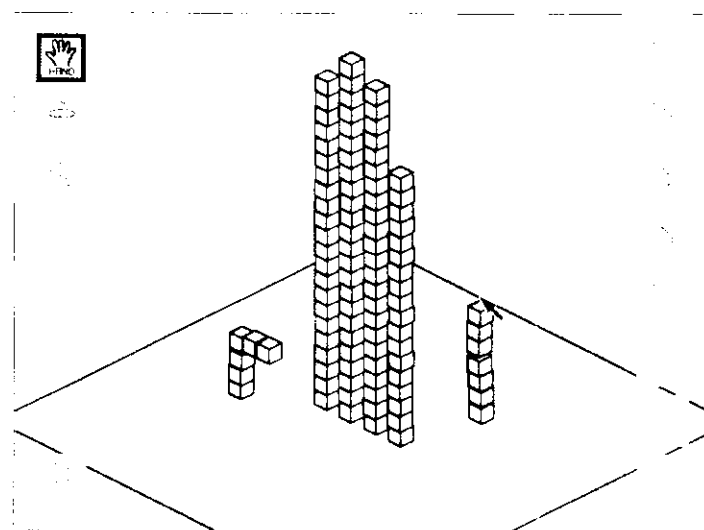


図 3.1 「Incompatible BLOCK」の画面

ツールが並ぶ。キャンバス内にはジオラマ箱の代わりに床オブジェクト(以下、床)を配置する。

「Incompatible BLOCK」の開発はMicrosoft社のVisual C++[32]と同社のMicrosoft DirectX 9SDK[33]を用いた。

前章の考察では、今後の展望に、高さの変更、立方体の移動、立方体の数の増減、ペンのサイズの変更を挙げた。本章では、これらの操作のための主観指向インタラクションデザインを考案した。

「はいぱーぺいんと」ではモードを切り替えた後、マウスクリックで立方体の追加や削除を行った。本研究では、これをマウสดラッグでモードの切り替えなしで行えるインタラクションデザインを考えた。マウสดラッグによる操作では、立方体が移動できることにより、ユーザは立方体の移動中の様子を見ることができる。

「はいぱーぺいんと」では透視投影を用いたが、ここでは平行投影法を用い、奥行きによらず常に立方体の見た目の大きさを変化させない方法をとった。これにより、ユーザは立方体の移動中に奥行きの変化が生じていることに気付かずにいる場合がある。この原理を使ってだまし絵のようにユーザを錯覚させる効果を得ることができると考えた。また、移動した立方体をユーザが期待する位置に配置することで有効な移動操作になるように、立方体の奥行きの設定は人間の心理特性を考慮して、描画イメージから得る主観的印象に基づくような配置となるようにした。これにより、楽しく操作でき且つ有効性も兼ね備えたインタラクションデザインとなることを期待した。本研究では、以上のようなだまし絵表現を用いて、高さの変更、立方体の移動、立方体の増減のための主観指向インタラクションデザインを考案する。ペンサイズの変更は、「はいぱーぺいんと」ではなかったズーム操作を用いる。3次元空間をズームしても、人間が見ているディスプレイのサイズは変わらない。このことを利用し、ズームによらず常に同じサイズのペンになるようにプログラムすることで、結果として3次元空間ではサイズが変更されるだまし絵のような主観指向インタラクションデザインを開発した[38-43]。

## 3.2 インタラクションデザイン

「Incompatible BLOCK」はモデリングや着色におけるペンのサイズに関する4つのだまし絵要素を用いたインタラクション表現を持つ。以下、それぞれの特徴を述べる。

### 3.2.1 立方体の移動

画面に対して「そのようになっているように見える」位置にマウสดラッグで移動した立方体は、移動先の状況から判断された結果、3次元空間内のユーザが期待する位置に配置される。

一般の3次元CGモデリングツール[29-31]では、マウスの移動距離とオブジェクトの移動距離は比例関係になるのに対し、「Incompatible BLOCK」の立方体移動操作ではそのような関係を成り立たせていない。立方体の最終的な3次元位置は立方体が置かれる状況によってのみ決定されるようにした。それにより、ユーザが期待する位置にすばやく配置可能となる。また、奥行きを意識せずに操作した結果が期待する位置に座標変換されることによりユーザ満足度の向上を図ることを考えた。具体的なインタラクション表現を以下に述べる。

- 1) 画面に対して床の上に移動させた立方体は床の上に置かれているように見え、立方体は3次元空間内の床の上に置かれる (図 3.2)

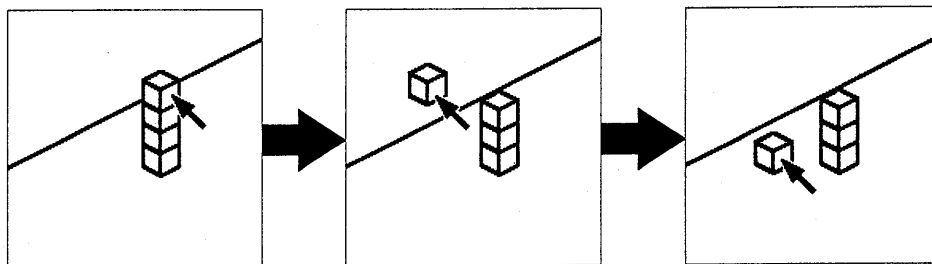


図 3.2 ブロックの床の上への移動

- 2) 立方体を画面に対して積み上がって見えるように移動させると、移動先にある立方体に積み重なるように立方体を配置することができる (図 3.3)

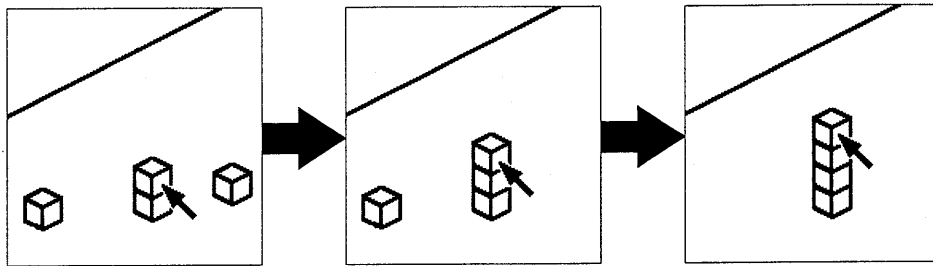
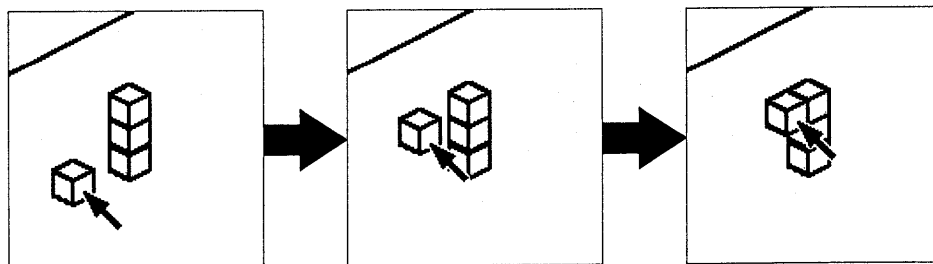
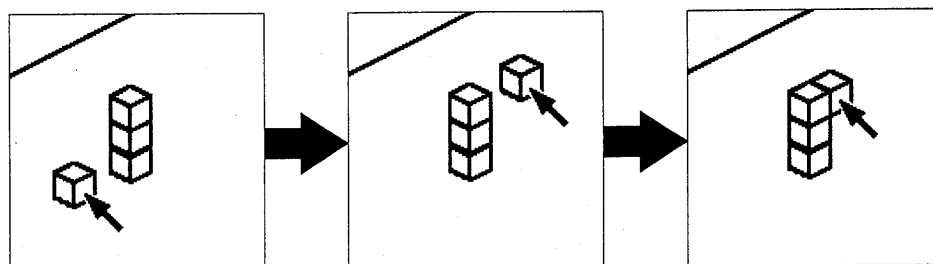


図 3.3 ブロックの積み上げ

このとき、移動した立方体は移動先にある立方体の前方 (図 3.4(a)) にも後方 (図 3.4(b)) にも配置できる。格子状に立方体を配置する傾向にある心理特性を考慮することで、移動する立方体と移動先にある立方体の位置



(a) 前方への配置



(b) 後方への配置

図 3.4 立方体への前後への配置

関係から自動的に最適な位置を決定するようにした。

また、床に接している形状に対して下方向から配置した立法体は、通常床の下に位置するために見えないと考えるが、配置した立方体は描画されるようにした（図 3.5）。これは見た目の位置を保つように形状全体が視点方向へシフトした結果である。操作していない形状が移動しているという意外性により、楽しく操作できることを狙った。

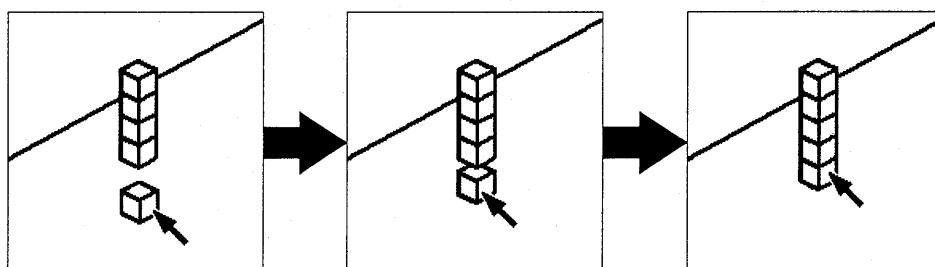


図 3.5 床に接する形状への下方向からの配置

3) 描画されない領域に置かれた立方体は、床に対して上下左右方向の移動が適用される（図 3.6）

立方体の見た目の位置はマウスマウスカーソル上にあるが、3次元空間内の位置は、移動開始前の位置を基準として床に対して上下左右方向に移動している。床の上が地面を意味するのに対し、描画されない領域は空中を意味するとし、ユーザは「宙に浮かせられる」と判断すると考えた結果である。

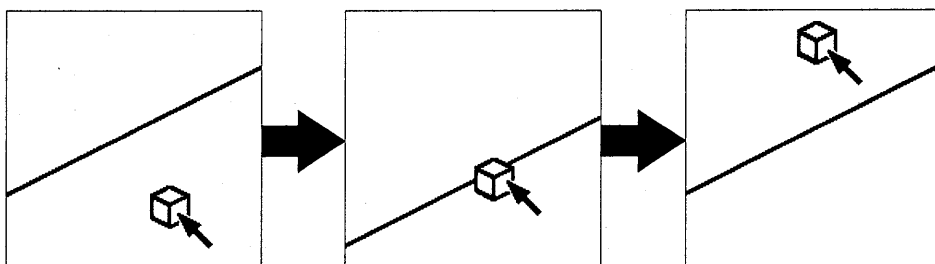


図 3.6 床に対して上下左右方向の移動が適用されるブロック

また、Y軸を3次元空間の上方方向から軸としたとき、図3.7のようにマウスドラッグでXZ平面に沿って床を動かすことが可能である。これにより、離れた場所にも床を移動して立方体を床の上に配置可能とした。また、視点を変えずに床を移動させて立方体の高さ位置を変更することもできる。

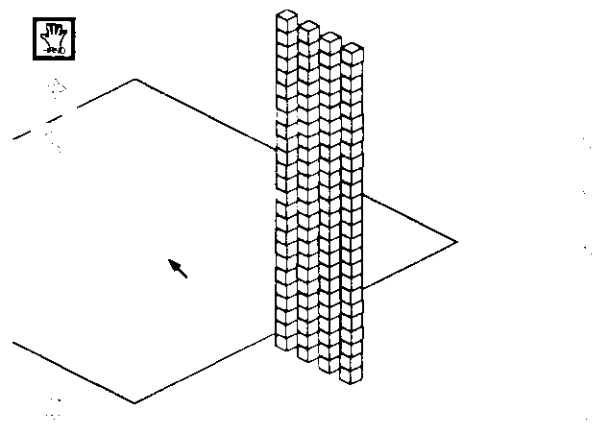


図 3.7 床の移動

### 3.2.2 影による立方体の高さ位置の変更

上記の法則の他に、「影」を用いた立方体の高さを変更する入力方法がある。影は立方体の底辺を下方向にマウスドラッグするとその立方体の影を引き出すことができる(図3.8)。その立方体は「浮いているように見える」ように、3次元空間内での立方体の高さ位置が変更される。立方体自体を触ることなく、立体の高さ位置は変更される。

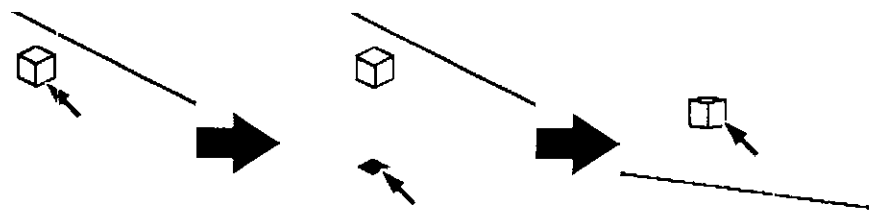


図 3.8 影の引き出しによる高さの変更



### 3.2.3 影による立方体の数の増減

前述の影を、一つの立方体から複数個引き出すことで、立方体を増加させることが可能である。実世界ではこれは視線方向に複数個の積み木が並んでいる状況に相当するように、3次元空間内にその影の数だけ立方体を増やすことができるようにした(図3.9(a))。反対に見かけが1つに見えるように視点変更し、複数の立方体が重なった時に少数の影を引き出すと、その数に合うように立方体を減らすことができるようにした(図3.9(b))。すなわち、立方体の数と影の数が対応するようにした。

これは3次元CGモデリングツールにおけるコピー機能と削除機能に相当するものであるが、3次元CGモデリングツールがメニューやアイコンの選択によりその状態を切り替えるのに対し、この立体増減操作は立方体を扱う手ツールで、切り替えることなしにコピー機能と削除機能を実行するものである。

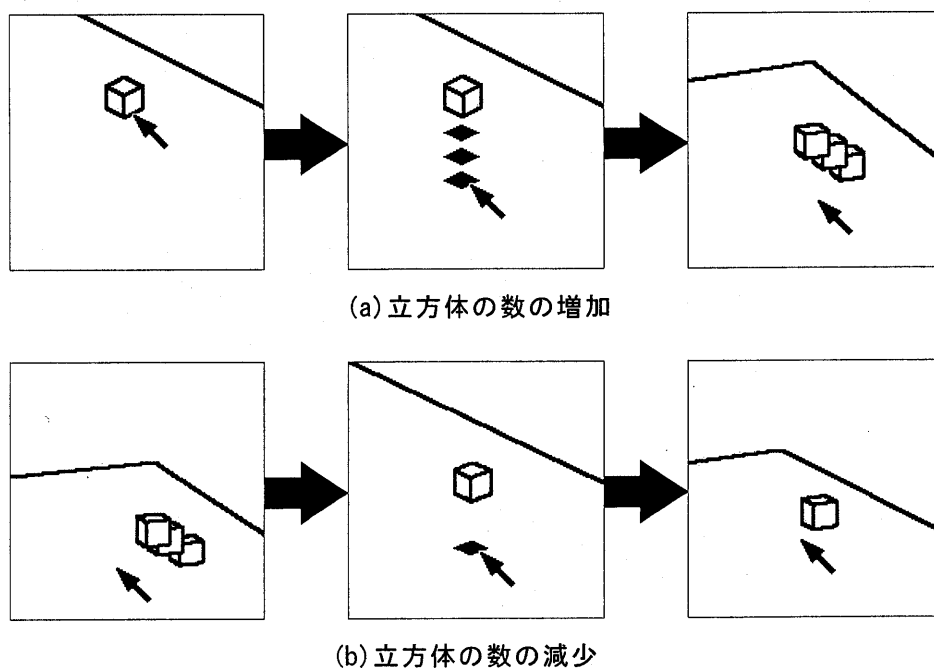
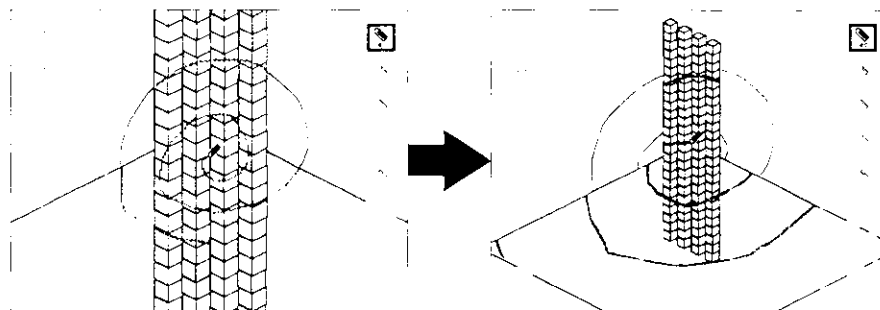


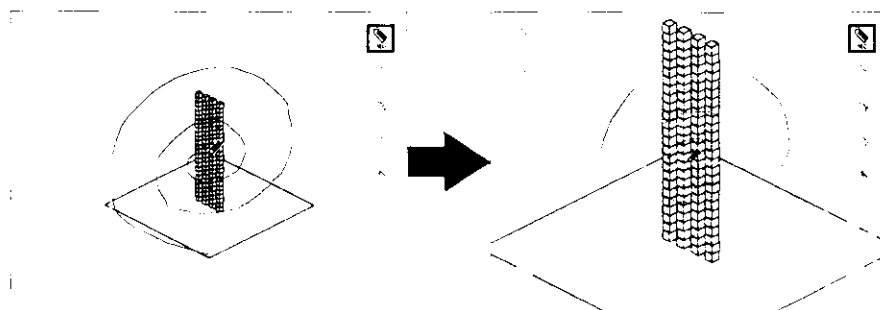
図 3.9 立方体の数の増減

### 3.2.4 ズームによるペンのサイズの変更

「はいはいーいんと」と同様、立方体、床、背景に、画面に対して線を描画することができるが、「Incompatible BLOCK」ではペンのサイズを変更できるようにした。描かれる線の「見かけ」の太さは空間をズームすることによって一定となる。すなわち、ズームアウトしているときはペンのサイズは太く(図 3.10(a))、ズームインしているときはペンのサイズは細くなるようにした(図 3.10(b))。背景に対するペンのサイズは、ズームに依存せず常に一定となるようにした。描面色は8色から選択する。立方体から引き出される「影」の黒色との混同を考慮して黒色は使用しないこととした。



(a)ズームインで太くなるペンのサイズ



(b)ズームアウトで細くなるペンのサイズ

図 3.10 ズームに依存するペンのサイズ

### 3.3 実装

本節では前節の特徴の実装方法について述べる。全体の処理の流れは「はいぱーぺいと」と同様であり、それについては「2.3.1 システム全体の処理の流れ」を参照されたい。また、以下の全ての処理はステップ 8)で行われる。

#### 3.3.1 立方体の移動

移動先の状況の判断は、2次元描画した移動対象である立方体と、床や移動先にある立方体との交差状態に応じてそれぞれの処理を行うようにした。交差状態の条件には以下の項目がある。

- 移動対象の立方体が床に重なる場合
- 移動対象の立方体が移動先にある立方体と重なる場合
- 立方体に何も重ならない場合

それぞれの実装方法について述べる。

##### 1) 移動対象の立方体が床に重なる場合

マウスイカーソル位置の3次元位置  $P_1$  から視線方向に向かうベクトル  $V_c$  と床の面  $F_f$  との交点位置  $P_2$  に立方体  $B$  を配置する (図 3.11)。

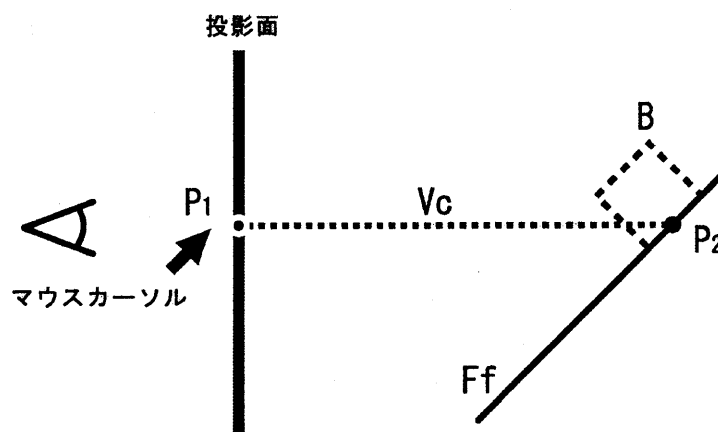


図 3.11 立方体の床への配置

2) 移動対象の立方体が移動先にある立方体と重なる場合

ユーザが格子状に立方体を配置しようとする心理特性を考慮することで、移動した立方体は移動先にある立方体に対して、それらの位置関係から最適な位置を決定するようにした。これを実現する手順を、図 3.12 を用いて説明する。

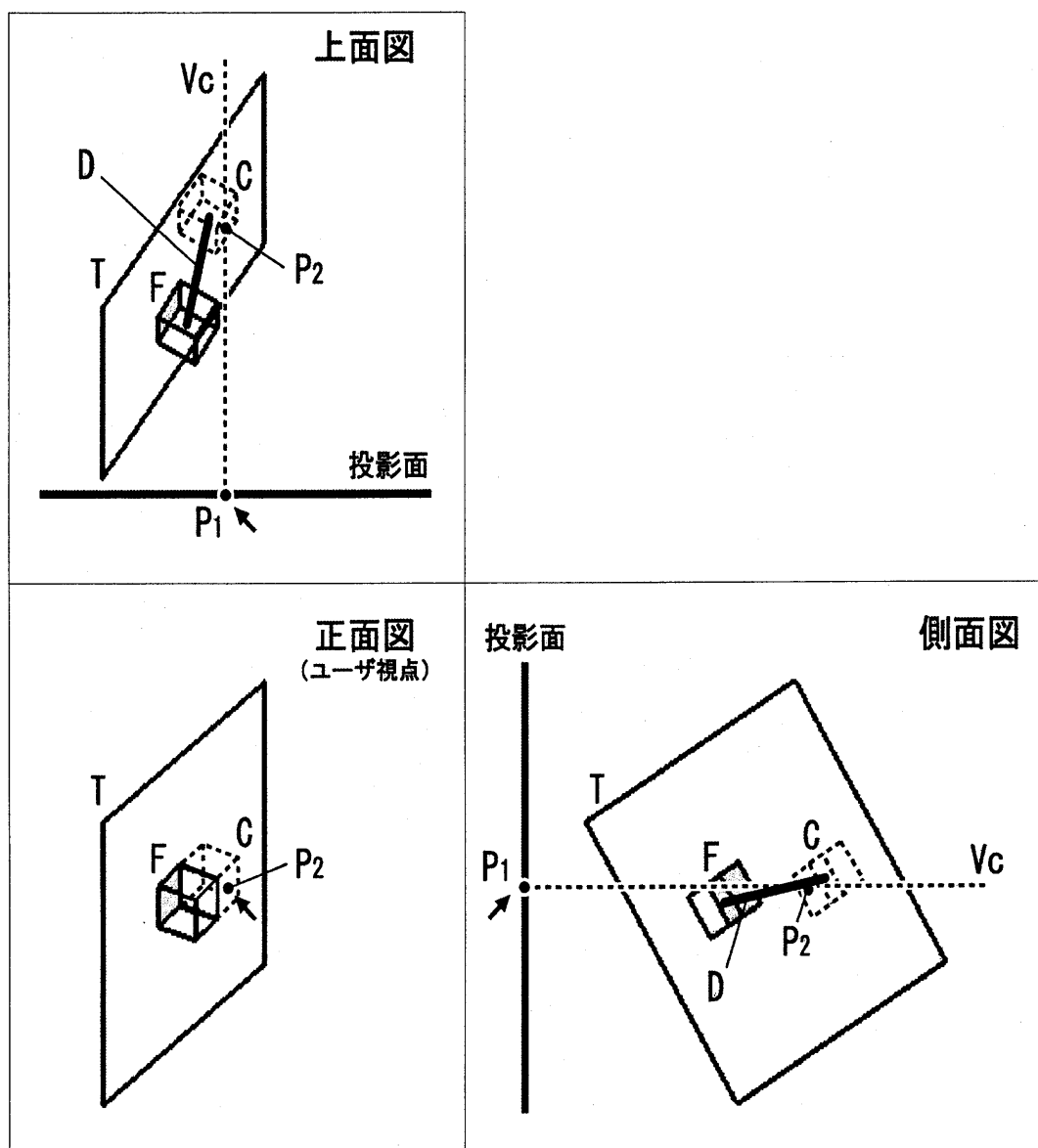


図 3.12 立方体の他の立方体への配置

1. 重なる対象の立方体の面 F (図中グレーの面) を含む無限平面 T とマウスマウスカーソル位置 P1 から視線方向に向かうベクトル  $V_c$  との交点 P2 を算出する。
2. P2 に仮想立方体 C の配置する。
3. 仮想立方体 C から重なる対象の立方体までの距離 D を算出する。
4. 1 から 3 の処理を立方体の持つ 6 面に対して行い、それぞれ D を求める。T と  $V_c$  の交点が求まらない場合は、D に限りなく大きい値を与える。
5. 最小の D を持つ仮想立方体の位置に移動対象である立方体を配置する。移動した結果が他の 3 次元立方体と衝突する場合は、次に近い面を用いる。
6. 5 で立方体が床の下にある場合は、その立方体を含む形状全体が床に載るように視点方向へシフトする (図 3.13)。この位置は条件 1 の床と重なる場合と同様で求められる。

重なる対象の立方体が複数ある場合は、すべての立方体に対して上記の処理を行う。また、重なる対象に床も含まれている場合は、立方体の処理を優先し、床の処理はしない。

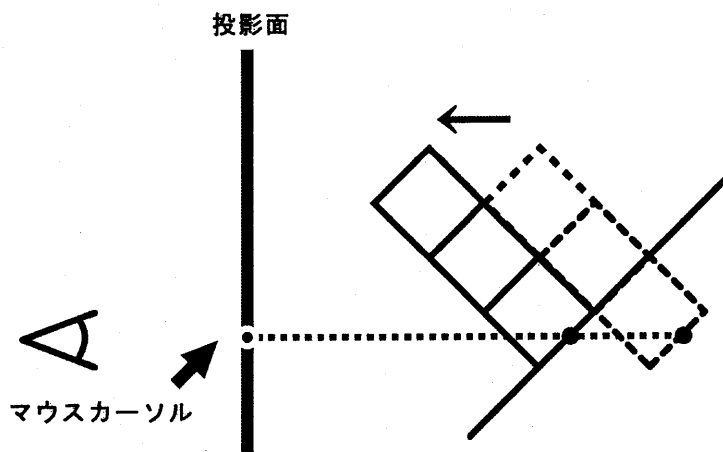


図 3.13 形状全体のシフト

### 3) 立方体が何も重ならない場合

移動開始前の立方体の中心を通り床に対して垂直な面  $T$  とマウスイカーソル位置の3次元位置  $P_1$  から視線方向に向かうベクトル  $V_c$  と交点位置  $P_2$  に立方体  $B$  を配置する (図 3.14)。

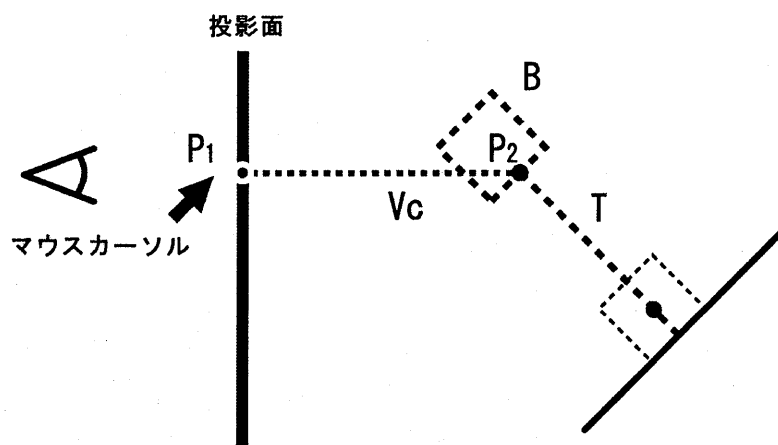


図 3.14 立方体の床への配置

### 3.3.2 影による立方体の高さ位置の変更

図 3.15 を用いて説明する。まず、マウスイカーソル位置の3次元位置  $P_1$  から視線方向に向かうベクトル  $V_c$  と床の面  $F_f$  との交点位置  $P_2$  を算出する。

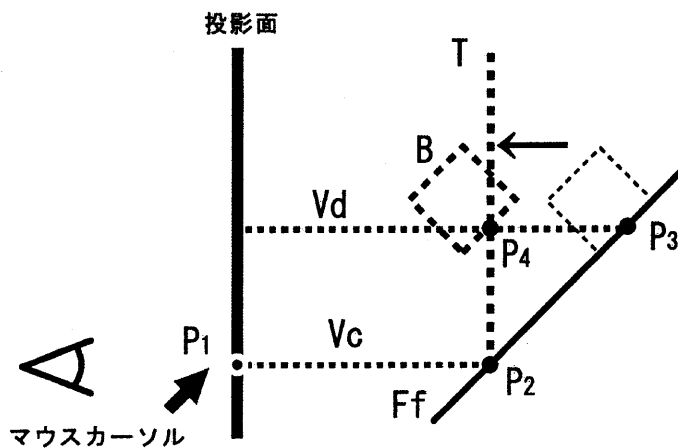


図 3.15 影による立方体の高さの変更

P2 を通りカメラビューに垂直な平面を T とおく。移動開始前の立方体の位置 P3 から視線方向に向かうベクトル  $V_d$  と平面 T との交点位置 P4 に立方体 B を配置する。このとき、P2 に影とする影オブジェクトを配置する。

### 3.3.3 影による立方体の数の増減

新しい影が引き出されると同時に、新しい立方体を生成するアルゴリズムとした。この時、新しく生成される立方体は前節の影による高さの変更で求めた位置に配置する。視点変更時は、影を持つ各立方体の中心のスクリーン座標を求め、これがほぼ一致する場合は影を持つ立方体を残し、他の立方体を削除するようにした。

### 3.3.4 ズームによるペンのサイズの変更

「Incompatible BLOCK」は平行投影で描画を行うため、ズームは形状の拡大縮小を用いることとした。もとの形状の大きさを変えて描画するアルゴリズムとした。ペンのサイズは、以下の拡大率の値に応じた条件分岐で、3段階のペンのサイズを切り替えるアルゴリズムとした。ペンによる着色の処理は「2.3.2 線描画」と同じである。

```
IF ズーム値が 0.3 以下 THEN
    ペンのサイズを 3 ピクセルにする
ELSE IF ズーム値が 0.6 以下 THEN
    ペンのサイズを 2 ピクセルにする
ELSE
    ペンのサイズを 1 ピクセルにする
```

### 3.4 評価

国内外の展示により、12歳から65歳までの年齢層の鑑賞者に「Incompatible BLOCK」を使用してもらい、行動観察及びインタビューにより有効性の検証を行った（図 3.16）[E5, E7-E9, E11-E14]。展示では1分弱の概要説明をした。立方体の高さ方向の移動はまだ意図しない結果となることがあるが、積み上げについてはほぼよく思った通りに立方体を配置できることが多かった。特に、置かれている立方体への背面方向への配置は3次元空間を意識せずに行っている様子が多く見られた。立方体の下方向からの配置は、原理を探ろうと試みる者が多かった。影による立方体の高さ位置の変更はあまり使用されなかったが、ハン サイズ変更は大きなサイズのハンで一気に塗りたい時に活用された。逸突したインタラクションの中では立方体の下方向からの配置と影による立方体の増減がもっともユーザーの興味を引いた。体験者の中に30分以上、「Incompatible BLOCK」の操作を楽しむ者もいた。インターネットで「Incompatible BLOCK」をダウンロードしたユーザーからは、立方体の配置が楽しいという意見が多い。制作した3次元形状を保存して続きを楽しみたい、という意見も得られた。



図 3.16 「Incompatible BLOCK」を使用する様子



また、「Incompatible BLOCK」をコンテストへ応募し、展示実演やTV番組放映による専門家の評価を得ることができた。以下に得られたコメントを引用する。

- 積み木という子ども向きのやさしい表現のなかに、数学的手法や錯覚させるトリックなど、機知に富んだ手法が取り込まれているのが楽しい [C6]。
- (要約) マウスの左ボタンと右ボタンの機能を変えたり、キーボードを押しながら操作するとモードが変わる機能で使いやすくして欲しい。3Dの世界をさらにその中に作って、自分のルールにのっとってソフトウェアを作っているのが楽しい [C5, C7]。
- (要約) 実際の積み木じゃ出来ないことがこの中では出来るところが面白い。いろんな可能性、いろんな未来が詰まっているソフトだと思った。やってみてすごく楽しかった [C5, C7]。
- 子供たちが空間感覚を身につけるためにあるような積み木という玩具をデジタルにした。なおかつ錯視、いわゆるイリュージョンまで体験できる余地を残している。よく考えられている。完成度が高いし可能性もある [C5, C7]。
- This sketch presents a system for free-form modeling in the vein of Sketch and Teddy and details the concerns unique to its block-based approach. This system seems promising; it will be interesting to see what creations emerge. (siggraph 2006 reviewer 1 の評価 [40, 41] )
- While I am curious about the real utility of the interface (what tasks is this actually suitable for) I find it an interesting interface, well described, fine to publish as is. (siggraph 2006 reviewer 3 の評価 [40, 41])

### 3.5 考察・まとめ

本研究では立方体配置の主観指向インタラクションデザインの改良とペンサイズの変更のためにズームを用いた主観指向インタラクションデザインを考案した。考案したインタラクションデザインを「Incomaptible BLOCK」のプロトタイプとして制作し、ユーザ評価や専門家による評価により検証した。

立方体の配置は、「はいぱーぺいと」ではマウスクリックで新規に追加するユーザインタフェースであったのに対し、「Incompatible BLOCK」ではマウสดラッグで立方体を移動するユーザインタフェースとした。「はいぱーぺいと」では追加するまで立方体は現れないが、「Incompatible BLOCK」では常に見える状態にあるため「はいぱーぺいと」に比べ位置の予測がつきやすくなった。また、これまでの3次元CGモデリングインタラクションでは成し得なかった視点変更を必要とせずにオブジェクトの背面方向への立方体の配置を可能としたため、目的とする位置にすばやく配置できるようになった。このようなアルゴリズムにより立方体の配置の操作性は向上したと考えることができる。しかし、立方体の高さの変更に関しては問題がみられた。この中で、立方体の下方向からの配置は原理を探ろうと試みる者が多かった。これは動かしている対象は動かないという人の心理特性を利用しただまし表現である。このような人の心理を裏返すだまし絵表現はユーザの興味を引くものとして有効なことを確認できた。立方体の下からの配置とは反対に、床に接する立体形状の最も下にある立方体を移動させても、その形状は床に接しているように見える通り、3次元空間内でシフトする表現を提案する意見もあった。

影による立方体の高さ位置の変更はあまり使用されなかったが、これは配置する高さが高くなるほど立方体は視点方向に移動するため、期待する位置での高さの調整が難しいためであると考えられる。立方体の下方向からの配置も同様に視点方向へ形状が移動するが、これが影による高さの変更に比べ

使用されていたのは、ユーザにとって形状がどこにあるかということよりも目的の形状が早く作れることを望むからであると推測できる。

影による立方体の増減は活用され、ユーザの興味を引いたインタラクションとなった。立方体の見た目は変更されないが、影の状況から判断した場合は立方体の数が増減していることは正しい。対象とするものは操作せずに、対象とするものの状態を変更できる表現は分かりやすさと意外性のバランスの取れたインタラクションとなることが発見できた。また、高さを変更するためのインタラクションが立方体の数を増やす操作として機能するように、1つのインタラクションから複数の機能を引き出す効果もユーザの好奇心向上に繋がるものと考えられることができる。

既存の3次元CGモデリングツールの使用経験があるユーザの方がそのような経験のないユーザに比べ、「見える通りに」操作した結果が期待する形状になっている事実に驚きを感じていたことは興味深い。

また、専門家の評価から、本主観指向インタラクションデザインが楽しい操作感をもたらしていることを確認できた。本研究で開発した「Incompatible BLOCK」に3次元CGモデリングツールとしての有用性があることも示された。「Incompatible BLOCK」の完成度は高く、教育方面や、その他、様々な分野への応用が期待されている。しかし、操作性に関して要求があるように、使いやすさの点で改善の余地がある。

今後の展望として、操作性の向上があげられる。本主観指向インタラクションデザインはだまし絵表現を行うが、容易にだまし絵表現を行えるインタフェースを考える必要がある。また、3次元CGモデリングソフトウェアには様々な機能が存在する。今後、主観指向インタラクションデザインとして開発を考えている機能をいくつかあげる。

- 回転、拡大機能の追加

立方体を回転、拡大できるようにする。

- グループ機能の追加

まとめて操作できるグループ機能を追加する。

- スイープ機能の追加

断面形状の移動の軌跡から形状を得るスイープ機能を追加する。

- 視点移動の追加

視点を投影面に対して上下左右に移動できるようにする。そして、これを用いた主観指向インタラクションデザインを考案する。

これらの機能のための主観指向インタラクションデザインを開発すると共に、操作性の考慮、様々な分野への応用を考えていきたい。

本章の終わりに「Incompatible BLOCK」を用いて制作されたモデルを図 3.15 に示す。

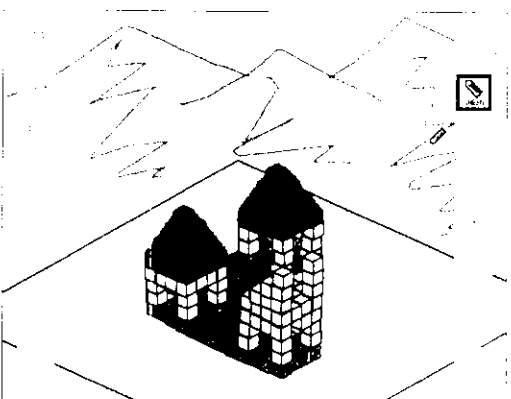
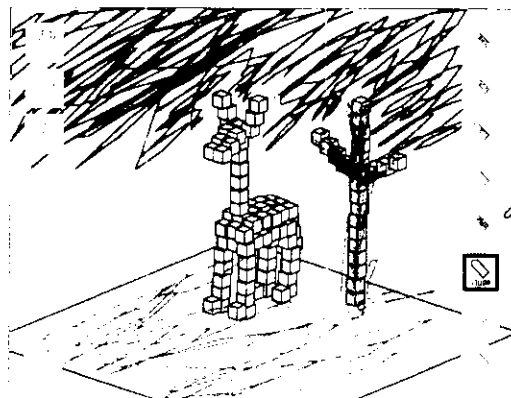
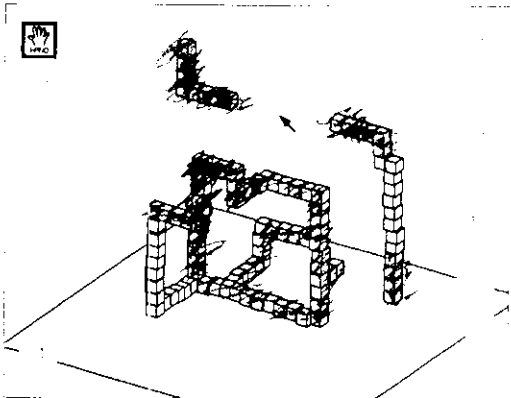
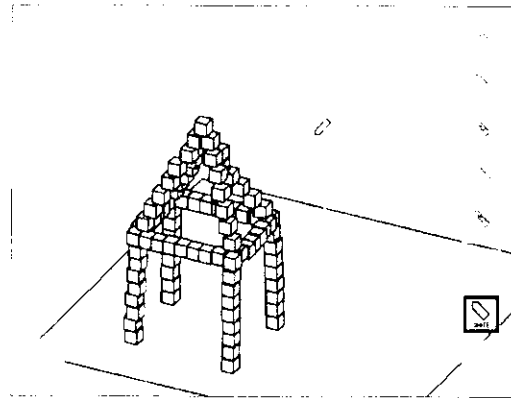
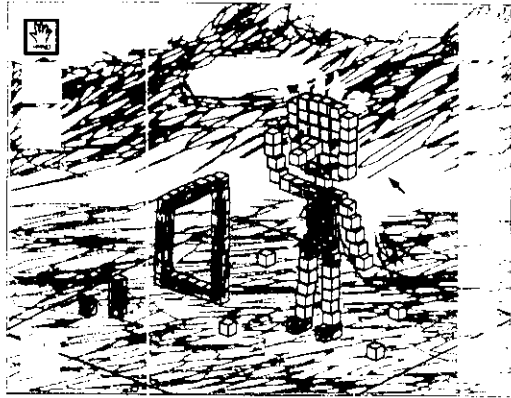


図 3.15 「Incompatible BLOCK」 の制作例