

CG の多視点表示による支援システム

石戸谷 樹* 五十嵐 悠紀*

概要. 本稿では x , y , z 方向それぞれからの 3 次元空間の表示やプログラム実行中のオブジェクトの移動, 回転を可能とする支援システムを提案する. 実行結果を表示する画面とは別の操作画面において, 生成された 3D シーンを x , y , z 方向それぞれから表示することにより, オブジェクトの位置関係などの把握を補助する. 更に, ドラッグ&ドロップによってオブジェクトを直接操作できることで位置の修正などを簡単にする.

1 はじめに

プログラミングによる 3DCG 制作ではオブジェクトの操作はすべて数値によって入力されるため, プログラムを実行しないと完成形はわからない. ゆえに, 座標空間の把握や完成形のイメージが要求される. しかし初心者の場合, これらの能力が十分でないために異なる位置にオブジェクトを配置してしまうことが考えられる. 加えて, 修正するにはコードを書き換えるごとに逐一実行結果を確認しながら作業する手間がかかってしまう.

そこで, 3 次元空間を多視点表示することで座標空間の把握を補助し, プログラム実行中のドラッグ&ドロップによるオブジェクト操作を可能とするシステムを提案する. 本システムは初心者が理解しやすい Processing[2]のライブラリとして実装した. 生成された 3D シーンを座標とともに多視点表示することで, 予期せぬところに配置されたオブジェクトも把握しやすくなる. x , y , z 方向それぞれの固定された視点で座標空間を表示することで移動および回転方向を絞り, 3 次元空間でのオブジェクトの操作をすることができる.

2 関連研究

GradeIV[1]は初心者が 3 次元 CG シーンのデバッグをしやすくするために, スクリーン上にレンダリングされる視覚情報とソースコード内の数値情報との関係を可視化するツールを提案している. 我々の提案システムは視覚情報と数値情報の関係を可視化する点では同一だが, オブジェクトの操作を数値情報に反映させることを可能とした.

また, Processing に備わっている Tweak Mode[3]

はプログラム実行中の数値変更が可能である. しかし, 数値を直接操作するために座標空間の把握が前提条件となる. 提案システムではオブジェクトを直接操作することで, 座標空間が把握できていない初心者にとってより扱いやすいものを目指す.

3 提案システム

3.1 x , y , z 方向からの 3 次元空間表示

提案システムは図 1 のように, 生成された 3D シーンを表示する画面と操作画面に分かれている. 操作画面ではキー操作によって図 2 のように x , y , z 方向からの視点を切り替えることができる. これにより, 一つのカメラでは死角になってしまうため, わかりにくいオブジェクトの位置関係も複数の視点から把握しやすくなる. この仕組みは 3DCG のモデリングソフトを参考にしており, ワールド座標系を視点ごとに回転させることによって実装している.

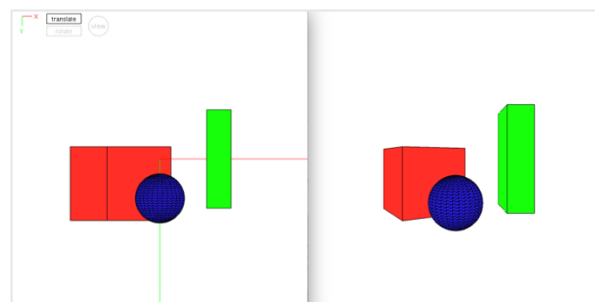


図 1. z 軸側から表示した操作画面と 3D シーンを表示画面

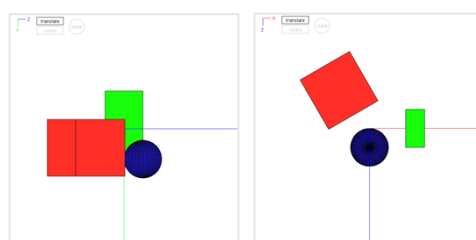


図 2. x , y 軸側からみた 3 次元空間

Copyright is held by the author(s).

* 明治大学 総合数理学部 先端メディアサイエンス学科

3.2 ドラッグ&ドロップによるオブジェクト操作

ドラッグ&ドロップによる3次元空間上でのオブジェクト操作は移動方向が一意に定まりにくい。しかし本システムでは、前述のように3次元空間を x, y, z 方向からの固定された視点で表示しているため、オブジェクトの移動する軸を2つに限定することができる。また、操作画面を平行投影にすることで、距離による見かけの大きさが変化しない。これらにより、3次元空間上でのドラッグ&ドロップによるオブジェクト操作を可能とした。オブジェクトの位置がずれていた場合、通常では座標の調整と実行を繰り返すことになる。しかし、ドラッグ&ドロップによるオブジェクト操作を可能としたことで、座標のズレの修正を一度で行うことができる。オブジェクト操作中は座標を常に表示することで、視覚情報と数値情報の関連づけをしている。

3.3 デバッグとしての機能

プログラミングによる3DCG制作では、入力ミスなどが原因で予期せぬ位置にオブジェクトが配置されてしまうことがある。しかし、カーソルキーで操作画面の拡大倍率を変更することにより、カメラ外に配置されてしまったオブジェクトの位置を把握することができる。

また、画面上のviewボタンを押すことで、画面およびコンソール上に座標が表示される(図3)。マウス操作によるオブジェクトの移動はコードに反映されないが、コンソールをみることでコードの書き換えもスムーズに行うことができる。

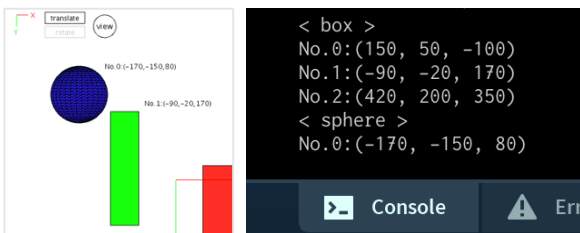


図 3. 実行画面とコンソールへの座標表示

4 結果と議論

本システムの有用性を確かめるためにユーザテストを実施した。テスト内容は、画像を参考に明治大学中野キャンパスの建物を対象に本システムを使用した場合と使用しない場合の2つの方法で作成してもらい、それぞれの時間を計測するというものである。被験者は普段CGに触れていない大学生5人であり、順序効果を防ぐため、テストの順番を被験者ごとに変更している。その結果を表1, 図4に示す。

表 1. ユーザテストの結果

被験者	先に実施したテスト	使用しない場合の時間	使用した場合の時間
1	使用しない場合	8分20秒	7分20秒
2	使用した場合	13分30秒	12分15秒
3	使用しない場合	14分34秒	10分05秒
4	使用した場合	14分01秒	7分54秒
5	使用しない場合	12分06秒	7分58秒

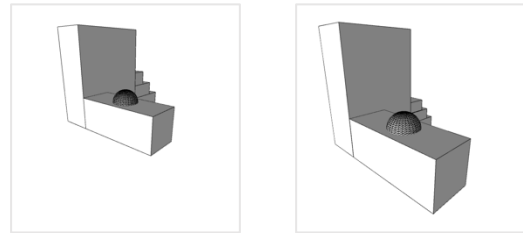


図 4. 被験者が作成した3Dシーン

ユーザテストの結果、被験者はテスト順序にかかわらず、本システムを使用した方が作成時間が短いというデータが得られた。被験者ごとに短縮された時間は異なるが、本システムは初心者がプログラミングで3DCG制作をする際の支援になると考えられる。被験者からは、「座標の調整がしやすい」「どこにオブジェクトがあるかが直感的にわかる」などの意見があった。一方で、「オブジェクトが重なってしまうと操作がしにくい」などの意見も挙がった。

5 まとめと今後の課題

本稿では、 x, y, z 方向からの3次元空間の表示やプログラム実行中のオブジェクト操作を可能とするシステムを提案した。座標空間が把握できていない初心者にとって数値のみでの3DCG制作は難しいが、多視点からのオブジェクトの視覚化や直接的な操作を可能とすることで、簡単に3DCGを扱える可能性を示した。今後は、オブジェクトのマウス判定を調整して重なった時の操作性を改善する。複雑なシーンの制作でも操作において不自由に感じないようにしたい。また、システム上で入力できるオブジェクトを増やすことで、制作の幅を広げることを検討している。

参考文献

- [1] Hidehiko Abe, Takeo Igarashi. GRADE-IV: visualizing graphics library operations in an executing program. In Proceeding SIGGRAPH 2007 posters, Article No. 118.
- [2] Processing. <https://processing.org/> (2018/7/25 確認)
- [3] Tweak Mode - Gal: portfolio (2018/07/25 確認) <http://galsasson.com/tweakmode/>