

# 教員の負担軽減を考慮した中学生対象 CG 作成支援システムの開発

佐藤貴之<sup>†</sup> 菊地卓也<sup>††</sup> 渡辺政和<sup>†††</sup>

中学生にプログラミングやCGに対する興味・関心を持ってもらうことを目的として、CG作成支援システムを開発している。本システムで扱うコンテンツは、中学校の教科「技術・家庭」の学習項目と適合し、システムが有用であることを確認している。しかし、システムの導入、運用、保守における教員の負担が大きいという問題が明らかになった。本論文では、リモートコンパイル技術を用いて、教員の負担軽減の観点からシステムを再構築する。さらに、構築した新システムを用いて授業形式のシステムテストを実施する。実施後の学習者アンケートの結果より、新システムは本研究の目的を達成する可能性が高いことがわかった。

## Development of Computer Graphic-Making Support System for Junior High School Students to Reduce Burden on Teachers

TAKAYUKI SATO<sup>†</sup> TAKUYA KIKUCHI<sup>††</sup>  
MASAKAZU WATANABE<sup>†††</sup>

We are developing a computer graphic (CG)-Making Support System with the objective of generating interest in programming and CG among junior high school students. The contents handled by this system are suitable for junior high school subjects studying, for example, technology and home economics, and we have confirmed that the system is useful for student learning. However, we discovered a problem in that the tasks that the teacher was responsible for, including system implementation, operation, and maintenance, created a large burden on the teachers. This research examines the use of remote compiling technology to reconstruct the system in order to reduce the burden on the teachers. In addition, we implement a classroom-style system test using the newly constructed system. The results of a student survey indicated there is a high probability that this new system will meet the objectives of the research project.

### 1. はじめに

プログラミングスキルを習得する場合、学習者は最初に数値、文字の取り扱いを繰り返し学習する方法が一般的である。しかし、数値、文字の入出力や処理に関する演習の繰り返しだけではプログラミング学習に飽きる可能性が高く、プログラムがうまく動作した時の達成感を持続しにくいと考えられる。そのため、ウィンドウ操作のようなグラフィカルな入出力を扱える高度なプログラミングスキルを得ようとする前に挫折してしまう学習者が多い。本研究では、その問題の解決を目指し、中学生の初学者でも短時間でグラフィカ

ルなスクリーンセーバを作成することができるCG作成支援システムを開発してきた[1]。本システムの目的は、簡単なCGプログラミング実習を通して、プログラミングやCG(コンピュータグラフィックス)に対する興味・関心と具体的なイメージを持ってもらうことである。さらに、本システムの内容は、中学校の教科「技術・家庭」の情報に関する技術[2]にある「デジタル作品の設計・製作」や「プログラミングに触れる」という学習項目に適合しており、中学校の情報教育での利用が期待できる。

ところが、本システムを用いた授業実施の準備段階で二つの問題点が明らかとなった。一つ目は学習者用PC全てにシステムをインストールするのは大変手間であること、二つ目はシステム全体のスペックが学習者用PCのスペックに大きく依存してしまい、スムーズな学習環境を提供できない可能性があることである。中学校の現場での利用を想定した場合、これらは準備

<sup>†</sup> 北九州市立大学  
The University of Kitakyushu  
<sup>††</sup> バイオニアシステムテクノロジー  
Pioneer System Technologies  
<sup>†††</sup> ブロードリーフ  
Broadleaf Co., Ltd.

する者の導入や運用に関する負荷が大きくなることを意味し、大変深刻な問題であると考えられる。

本論文では、導入や運用保守に関する問題を解決するため、リモートコンパイル技術を用いてクライアントサーバ方式で既存のシステムを再構築する。これにより、学習者用 PC にシステムをインストールする作業が必要なくなり、保守の対象がサーバに限られ、導入や保守の負荷が軽減される。

さらに、本論文では、新システムを用いて中学生を対象とした授業形式のシステムテストを実施し、その結果をまとめる。授業終了後に行った学習者アンケートの結果から、学習者に CG やプログラミング技術の具体的なイメージと興味を持たせられることがわかり、本システムによる教育効果が得られることを示唆する結果となった。

本論文の構成は以下の通りである。2 章でシステムの開発方針を述べた後、システム用プログラミング言語 CGML と CGML で記述したプログラムから CG を作成する内部処理の流れを示す。3 章ではリモートコンパイル技術による新システムとそれを用いた学習者の利用の流れについて述べる。4 章では、中学生を対象とした授業によるシステムテストの実施内容とアンケート結果を示し、5 章で本論文をまとめ、今後の課題を示す。

## 2. システムの概要

本章では、システムの開発方針と中学生に理解しやすいプログラミング言語を目指した CGML の概略を述べ、CGML から CG やスクリーンセーブを作成する内部処理動作について示す。

### 2.1 開発方針

本研究の目的は、CG プログラミング実習を通して、中学生の初学者に CG やプログラミングの具体的なイメージを持たせ、興味・関心を高めることである。そこで、本研究では、以下の方針に基づいてシステムを開発する。

- 学生の集中できる時間を考慮し、合計3時間で完結できる実習内容とする
- プログラミングは「触れる」程度にとどめ、実習による課題の達成感を得ることを優先する
- 学習者が自宅で使用できるオリジナルスクリーンセーブを学習成果物として提供する

### 2.2 システム用プログラミング言語 CGML

CG 作成で頻繁に用いられている OpenGL (Open Graphics Library) や GLUT (OpenGL Utility Toolkit) の開発では、学習者は C 言語の基礎習得が前提となり、

かつ、空間図形の移動を表現する行列などの数学的知識が必要になる。2.1 で述べた目的や開発方針と照らし合わせた時、このままでは中学生の初学者に CG 作成実習をさせることは困難である。そこで、中学生でも短時間で理解できる単純なプログラミング言語 CGML を開発した。ドリトル[3]、POV-ray[4]、Solid Interpreter[5]を参考に、以下の方針に基づいて言語を設計する。

#### ● 学習者が読みやすい言語

中学生でも理解できるプログラミング言語にすることで、プログラミングに対する親しみやすさとバグやエラーの修正の容易さが向上すると考えられる。CGML では、プログラミング言語に一般的に用いられている英語と日常使用している日本語の間に位置付けられる訓令式ローマ字表記を使用し、命令名を見て何を行うものかを直観的に理解できるように配慮する。

#### ● 学習者が作りやすい言語

本システムでは、スクリーンセーブ作成のようなグラフィカルな操作を可能にするプログラミングスキルを短時間で学習者に習得させる必要がある。CGML では、図形ごとに性質を設定し、シーン (全体の情景) を構成する方法を採用する。これはキャンバスに基本的な図形を一つずつ描画するというを意味し、中学生にも馴染みのある絵画の制作に似ている概念である。シーン内のそれぞれの図形に対し、モデル、形状、場所、色、動作の性質を付加することで図形の配置、描画、アニメーションを実現する。性質の付加はそれぞれの図形に対して性質を代入するイメージになると考え、「←」(実際のプログラミングでは<)を用いる。さらに、CGML の記述は全て「名前 ← 命令(値);」という形式に統一する。なぜなら、複数の演算子を提供すると、学習者がプログラミング実習時に混乱をきたす可能性が高いと考えたからである。同じ理由により、CGML では、出来る限り命令の種類を少なくし、単命令で結果が変わる形式とする。

設計した CGML の主な命令を表 1 に示す。

表 1 CGML の主な命令

図形の作成	tukuru
中心座標	iti2d, iti3d
X,Y,Z 軸の傾き	katamuki2d, katamuki3d
色彩	iro
モデル	katati
動作	idou2d, idou3d, ziten, kouten

表1において、`tukuru` で取ることのできる値は、正多角形 (`takakukei`)、楕円 (`en`)、卵型 (`tamago2d`)、直線 (`sen`)、直方体 (`tyokuhoutai`)、球 (`kyu`)、多角柱 (`tyu`)、多角錐 (`sui`) である。表1の命令群を用いて作成したプログラム例を表2に示す。

表2 CGMLプログラム例

```
sankaku <- tukuru(takakukei);
sankaku <- tyouten(3);
sankaku <- katati(sen);
sankaku <- iro(siro);
sankaku <- idou3d(1,2,-1);
```

表2では、一番上の行で `sankaku` という名前の多角形を定義する。2行目から4行目では、頂点数は3である、線で描かれている、色は白であるという3つの性質を順に `sankaku` に付加する命令が記述されている。

表2の一番下の行の `idou3d(1,2,-1)` は、1単位時間に  $x$  軸方向に1、 $y$  軸方向に2、 $z$  軸方向に-1移動することを意味している。1単位時間とは、現在表示されている画像から次に表示させる画像までの再描画時間を指し、スクリーンセーバの場合は40m秒である。

本システムの描画においてはOpenGLの描画領域を用い、描画範囲は各軸[-10,10]としている。本システムでは、描画範囲外に出た図形は同じ軸の反対の座標から再び現れる仕様としている。例えば、物体が(-6,10,4)から  $y$  軸に正の方向に移動した場合、(-6,-10,4)から出現する。表2のプログラムをコンパイルし、実行したCGを図1に示す。図1は白い三角形が時間の経過と共に(1,2,-1)方向に移動しているCGである。



図1 表2のプログラムを実行したCG

### 2.3 システムのコンパイル処理の流れ

学習者が入力したプログラムをCGとして表示できるようにするシステム内部の処理の流れを以下に示す。

最初に学習者がシステムに入力した文字列をGLUTに基づくC++プログラムに変換し、C++プログラムファイル(.cpp)として保存する。保存したC++プログラムファイルをもとに、C++コンパイラと環境設定ファイル(実行形式ファイル用、または、スクリーンセ

ーバファイル用)を組み合わせてコンパイルを行い、実行形式ファイル(.exe)、または、Windows用スクリーンセーバファイル(.scr)を作成する。そのため、本システムはOSがWindowsであるクライアントのみの対応となっている。

## 3. リモートコンパイル技術を用いたシステムの構築

本章では、既存システムの導入、運用、保守の観点からの問題点を挙げ、それを解決するための新しいシステムを提案する。具体的には、リモートコンパイル技術を用いて、クライアントサーバ方式により既存のシステムを再構築する。

### 3.1 既存システムの問題点と変更によるメリット

既存システムを利用して実習を行うには、学習者用PC全てにシステムをインストールしなければならない。しかし、中学校のパソコン演習室で1クラス40台程度のPCにシステムをインストールする手間は想像以上に大きい。また、システムにバグがあった場合やバージョンアップしたシステムの導入も、新規導入と変わらない程度の負担となる。学習指導、生活指導で多忙な現職教員の立場でシステムの導入、運用、保守に対する負荷を考えなければ、新しい教材、システムを構築しても、広く中学校に普及しないと考えられる。そこで、本論文では、リモートコンパイル技術を用いたクライアントサーバシステムを提案する。

新しいシステムの概略を図2に示す。学習者(利用者)はクライアントPCからWeb画面を通してCGMLでプログラムを記述し、プログラムファイルをサーバに送る。サーバはクライアントから送られたプログラムを2.3で述べた流れでコンパイルし、コンパイル結果の実行形式ファイルをサーバ自身に保存する。学習者は実行形式ファイルをサーバからダウンロードすることにより、自分が作成したCGを見ることができる。スクリーンセーバ作成もこれと同様の流れとする。

クライアントPCに求められる条件は、OSがWindowsでありネットワークに接続していること、Webブラウザがインストールされていることである。

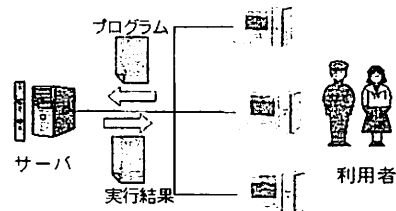


図2 新システムのイメージ

提案する新システムのメリットは以下が挙げられる。

- クライアント PC にインストールや新たな設定が不要である
- 保守はサーバ 1 台のみに集中すればよい
- システムはクライアント PC のスペックにほとんど依存しない
- サーバを学外に公開すれば、家庭（学校外）学習にも利用できる

すなわち、これらは学習者のメリットが大きいだけでなく、授業を準備する教員側の導入、運用保守の負担が劇的に減少することを意味する。

### 3.2 システム利用の流れ

本システムにおける学習者の CG 作成の流れは次の通りである。

まず、学習者は自分のクライアント PC の Web ブラウザを開く。次に、サーバの URL を Web ブラウザに入力し、表示されるログイン画面に ID・パスワードを入力してログインすると、図 3 に示すインターフェースが表示される。図 3 の左側には CG 作成の方法、原理を学習するために 1) で開発したテキストを Web 用に編集しなおした Web テキスト画面がある。右側にはプログラム名を入力するボックス、プログラムを入力するボックス、コンパイルボタン、スクリーンセーブ作成ボタンを配置したプログラム入力画面がある。Web テキストとプログラム入力画面を一つにすることで、学習者の視線移動が少なくなり、目が疲れにくくなることを目指す。

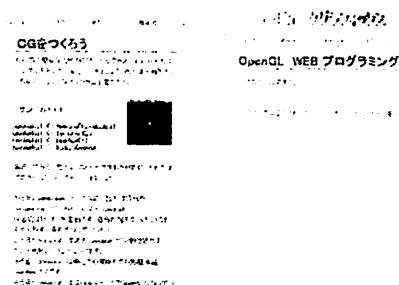


図 3 新システムのインターフェース

プログラム入力画面にプログラム名、プログラムを入力したら、画面下部にあるコンパイルボタンを押す。すると、コンパイル結果がプログラム入力画面上部に現れる。プログラムにエラーが含まれる場合は、図 4 に示すようなエラーが上部に表示される。

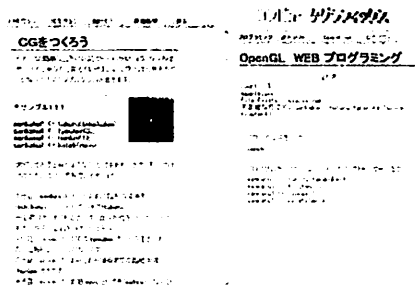


図 4 エラー画面（右上部）

正常にコンパイルした後、画面上部にあるダウンロード画面遷移ボタンを押してダウンロード画面へ行き、ダウンロード画面から表示したい CG の実行ファイル名をクリックする。すると、実行形式ファイルと glut32.dll の二つのファイルを一つに圧縮した zip 形式のファイルをダウンロードできる。右側にダウンロード画面を表示しているインターフェースを図 5 に示す。

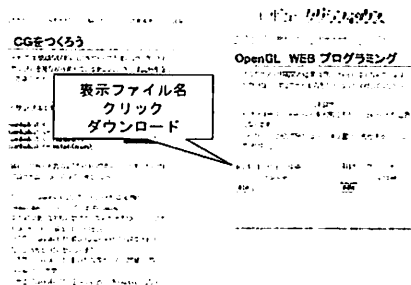


図 5 ダウンロード画面（右側）

zip 形式のファイルをダウンロードし、それをダブルクリックして解凍した後、フォルダ内にある実行ファイルをダブルクリックすることにより、自分が作成した CG を表示することができる。スクリーンセーブを作成する場合、図 3 でコンパイルボタンを押す代わりにスクリーンセーブ作成ボタンを押すこと以外は、CG 作成時と同じ流れで行う。

## 4. 授業形式によるシステムテスト

提案した新システムを構築後、授業形式のシステムテストを実施した。本章では、まず、授業の流れ、概要、教室環境を説明し、その後、学習者のアンケート結果を示し、授業の様子を撮影した動画により本授業の問題点を挙げる。

#### 4.1 授業のスケジュール

本システムの目的は、3時間以内の実習でCGやプログラミングを具体的にイメージしてもらい、かつ、それらに興味関心を持ってもらうことである。また、学習者の到達目標は「オリジナルスクリーンセーバ」を製作することとする。これらの要件を満たすように設計した授業のタイムテーブルを表3に示す。

表3 授業のタイムテーブル

オリエンテーション	10分
図形を描いてみよう	50分
休憩	5分
動かしてみよう	30分
休憩	10分
スクリーンセーバを作ろう (作り方説明)	10分
スクリーンセーバを作ろう (製作)	50分
作品発表	15分

授業の流れは、最初にシステムの使用方法を説明しながら静止面のCG作成を行い、次に動画CG作成、最後に学習者が決めたテーマによるスクリーンセーバ製作実習とする。授業の最後に作品発表を行い、他の学習者の成果を見ることで刺激を与え、お互いのレベル向上のきっかけ作りを狙う。

#### 4.2 授業の実施内容

授業の実施内容は以下の通りである。

- 期日：2008年11月29日(土) 13:00~16:00
- 講師群人数：4名  
(メイン講師：1名、サブ講師：3名)
- 学習者数：6名
- PC環境：学習者1人にWindows XPのPCI台  
講師群はメイン講師とサブ講師からなり、メイン講師は、主としてWebテキストをスクリーンに投影しながらの全体説明と授業進行の統括を行う。サブ講師は、

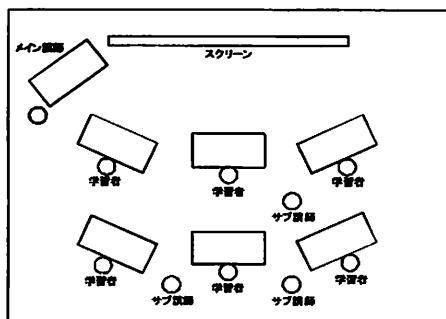


図6 授業時の学習環境

実習中の学習者の質問に応じて個別にディスカッションをする役割とする。学習者は、公開講座としてWebサイトから募集をかけて、自ら応募してきた中学生である。学習者、講師群の配置を含んだ授業当日の学習環境を図6に示す。

システムテストで用いたサーバのスペックは、以下の通りである。

- OS：Windows Server 2003 Enterprise edition SP2
- CPU：AMD Athlon 64 Processor 3500+ 2.20GHz
- メモリ：512MB RAM

授業当日のCG作成実習の様子を写真1、2に示す。これらは共にスクリーンセーバを作ろう(製作)の時の様子を撮影したものである。写真1の前方のスクリーンには、本システムで作成できるCGの例を示している。このCGはサブ講師がシステムテストの前に作成したものである。

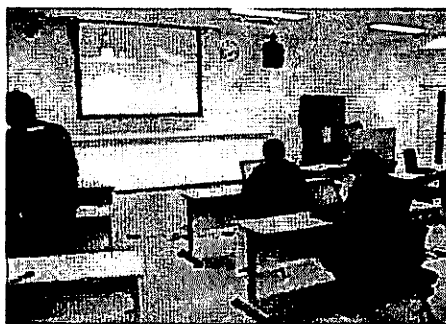


写真1 スクリーンセーバ作成実習の様子

写真2は、椅子に座っている学習者と立っているサブ講師とのディスカッションの様子を撮影したものである。

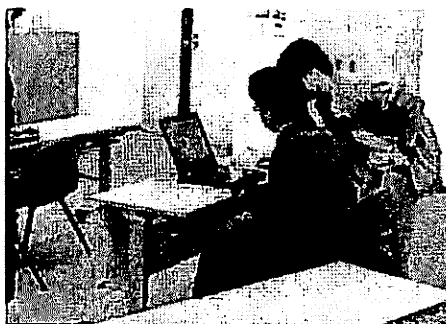


写真2 サブ講師とのディスカッション

実際の授業では、コンパイルエラーの修正と同じ図形を複数作成することに関する質問がほとんどであった。コンパイルのエラーでは、プログラムの末尾にセミコロン（ ; ）を付け忘れるミスがほとんどであった。複数の同じ図形の作成では、ほとんど同じ性質の図形を二つ作成するのに、別々の名前を付けて区別するという概念がわかりにくかったようである。この原因として、Webテキストでは一つの図形を操作する方法のみ記述しており、複数の図形を操作する方法が明示的に書かれていなかったことが考えられる。

#### 4.3 学習者の授業アンケートと動画による授業の分析

本授業の終わりに、学習者からアンケートを採取した。アンケートの質問を表4に示し、学習者の回答を表5に示す。回答は4段階評価とし、4が最も肯定的な意見、1が最も否定的な意見とする。

質問1から、未経験の中学生が4名集まり、授業の進捗、難易度などを解析することができることがわかる。質問5と7から、プログラミングに興味を持っている中学生にとっては、本授業がとても楽しいものであることがわかる。さらに、質問3、4、6から、CGやプログラミングに対し以前よりも興味、関心を強く持ち、より具体的なイメージを持ってもらえたことがわかった。また、質問7と8から、課題は難しいながらも楽しんで実習している状況がわかる。質問9で資料に対して肯定的な意見が多かったものの、質問10で授業時間が少なかったと感じる学習者もいた。

表4 授業アンケートの質問

1. 以前にプログラミングに触れたことはありますか
2. プログラミングに対してどのくらい興味、関心がありますか
3. 以前に比べてプログラミングに対する具体的なイメージは持てましたか
4. 以前に比べてプログラミングに対する興味、関心は増えましたか
5. CGを作るのは楽しかったですか
6. 以前に比べてCGに対する具体的なイメージが持てましたか
7. プログラミングは楽しかったですか
8. プログラミングは難しかったですか
9. 資料はわかりやすかったですか
10. 各段階で十分な学習時間は取れましたか

表5 授業アンケートの回答

質問番号	4	3	2	1	未記入
1	1	1	0	4	0
2	2	4	0	0	0
3	2	4	0	0	0
4	1	4	0	0	1
5	5	1	0	0	0
6	2	4	0	0	0
7	4	2	0	0	0
8	0	2	4	0	0
9	2	4	0	0	0
10	2	2	2	0	0

次に、授業中の様子を動画撮影したものを、本システムの開発、システムテスト、公開講座に全く関わらなかった高専学生5名に見てもらい、意見を収集した。それらの意見を表6に示す。

表6 授業の様子を見た意見

良かった点	<ul style="list-style-type: none"> <li>・説明がわかりやすい</li> <li>・学習者の操作画面と同じ画面をスクリーンに投影しての解説がわかりやすい</li> <li>・動作の説明の前に、サンプルを表示して最終的にどのくらいのものができるかを提示したのはよい</li> <li>・学習者の不具合をサブ講師が回って解消するのが良い</li> </ul>
悪かった点	<ul style="list-style-type: none"> <li>・もう少し説明を簡略化しても良い</li> <li>・解説の時間が長く、学習者が何もしない時間が多いと感じる</li> <li>・受講者から質問しやすい工夫があると良い</li> <li>・学習者同士のディスカッションを許容できるような環境を提供するとよい</li> </ul>

悪かった点にある解説の時間が長いという指摘と学習者アンケートの質問10の回答結果より、解説をできる限り最小限に絞りPCに触れる時間を多く取るように工夫する必要があることがわかる。例えば、解説をしながらメイン講師と同時にPCの操作をさせるという工夫が考えられる。

良かった点の中でサブ講師が学習指導を効果的にしていると述べているが、実際の中学校において40人程度の演習室で授業すると仮定した場合、講師群の数は多くは望めない。そのため、学習者同士が教えあうことで演習できる環境を作ることが必要であると考えら

れる。

授業で用いたシステムでは、学習者がCGを閲覧するには、実行形式ファイルやスクリーンセーブファイルと共にglut32.dllが必要であった。そのため、学習者が作成したCGをダウンロードした後、zip形式のファイルを解凍する操作が必要となった。さらに、何度もダウンロードしているうちにデスクトップにzip形式のファイルと解凍後のフォルダが多く並ぶ現象が起こった。授業終了後、入力された文字列をC++プログラムとして保存する時にGLUT関連のヘッダファイルの記述内容を挿入することで、glut32.dllなしでCGを閲覧することを可能にし、システムを改良した。

## 5. おわりに

本論文では、中学生対象CG作成支援システムを教員の負担を軽減する観点から、クライアントサーバ方式によるシステムを提案しシステムを再構築した。新システムを用いた教育実践の結果から、学習者にCGやプログラミングに対する具体的なイメージを持たせ、関心を高める傾向にあると予想でき、本システムの有用性が示唆できる。しかし、検討しなければならない問題点が存在する。

一つ目は、CGMLの言語設計に関する問題である。CGMLでは、重要なプログラムの概念である条件分岐や繰返しを除外していることをはじめとして、プログラミングの本質に関わる機能をあえて実装していない。まずは情報技術に興味を持たせることが大切であるという考えによる設計方針であるが、この言語仕様のままで実際の中学校で教科教育として利用できるかを検討する必要がある。これに関連して、実際の中学校の教科教育の一環として利用する場合は、授業時間の配分をはじめとしてカリキュラムを再検討する必要がある。また、今回の教育実践の規模が小さかったため、実際の中学校では、興味関心や性格により学生の取り組み方にばらつきが出てくる可能性もある。そのあたりも考慮した授業設計を考えていく必要がある。

二つ目はシステムに関連する問題である。システムテスト実施にあたり、サーバの負荷実験を行っているものの実際の現場でシステムテストをしていないことなどの課題が残っている。また、システムのインタフェース関連では、中学生に対してわかりにくいエラー表示となっているため、今後検討が必要な部分である。

より広く中学校で利用してもらえよう、なるべく早い段階でシステムテストに協力頂ける中学校と連携し、授業展開をしたいと考えている。

**謝辞** システムテスト実施に多大なご協力を頂いた仙台電波工業高等専門学校企画係のみなさまに謹んで感謝の意を表する。

## 参考文献

- 1) 菊地卓也, 佐藤貴之:中学生対象コンピュータグラフィックス作成支援システムの開発と教育実践, 第7回情報科学技術フォーラム(査読付き論文), 第4分冊, pp. 59-60(2008).
- 2) 文部科学省新しい学習指導要領「生きる力」中学生学習指導要領「技術・家庭」, [http://www.mext.go.jp/a\\_menu/shotou/new-cs/youryou/chu/gika.htm](http://www.mext.go.jp/a_menu/shotou/new-cs/youryou/chu/gika.htm)
- 3) 兼宗進, 中谷多哉子, 御手洗理英, 福井眞吾, 久野滄: 初中等教育におけるオブジェクト指向プログラミングの実践と評価, 情報処理学会論文誌, Vol. 44, pp. 58-71 (2003).
- 4) POV-Ray-The Persistence of Vision Ray tracer, <http://www.povray.org/>
- 5) 新津滄:教育用3次元ソリッドモデラーの開発, 平成13年度経済産業省未踏ソフトウェア創造事業報告書(2002).