

# 「ものづくり」を取り入れた プログラミング教育システムの提案

田中 元気<sup>1</sup> 谷口 航平<sup>1</sup> 桑原 宏輔<sup>1</sup> 葛尾 耕平<sup>1</sup> 安達 拓也<sup>2</sup> 濱川 礼<sup>2</sup>

**概要:** 本論文では、3D スキャナを用いた「ものづくり」要素を取り入れたプログラミング教育システムの開発と児童への評価実験の結果について述べる。昨今、各国でプログラミング教育を推進する活動が見られ、児童に向けた様々なプログラミング教材が開発されているが、児童の学習意欲を引き出し興味をもたせることは難しい。そこで我々は現実世界での「ものづくり」も取り入れることで児童の関心を引くことができると考えた。本研究では、与えられた課題をユーザが現実世界での「ものづくり」と仮想世界での「プログラミング」によって解決していくシステムを開発した。ユーザは現実世界で「レゴブロックなどで自作したオブジェクト」を3D スキャナでスキャンし、それをPC上で3Dモデルに変換する。その後、仮想世界で変換された3Dモデルに「プログラミング」を施すことで動作を制御し、与えられた課題に取り組む。提案システムでは「ものづくり」と「プログラミング」二つの面から創意工夫ができるため、児童の関心を引くことができる。本システムの評価実験を児童に対して行ったところ、Scratchと同等以上の評価を得ることができた。

## Proposal of programming educational system with manufacturing technology

GENKI TANAKA<sup>1</sup> KOUHEI TANIGUCHI<sup>1</sup> KOUSUKE KUWABARA<sup>1</sup> KOUHEI KAZUO<sup>1</sup>  
TAKUYA ADACHI<sup>2</sup> REI HAMAKAWA<sup>2</sup>

### 1. はじめに

昨今、プログラミング等のIT教育を推進する活動が各国で見られる。これは今後IT関連のビジネスが拡大していくと予想される一方で、それに対応するIT人材の数が追いつかないと予測されているためと考えられる。

2014年の文部科学省による調査報告書[1]より各国の活動の例を挙げる。イギリスでは教育課程の国家基準として導入されているナショナルカリキュラムにおいて、従来の教科「ICT」に代わり教科「Computing」が新設された。教科「Computing」は義務教育の5~16歳で必修となり、2014年9月より実施されている。

アメリカでは、国全体でプログラミング教育を推進している。例えば、NPO法人のCode.org[2]はプログラミング教育を推進する活動を行っており、オバマ前大統領も自ら

プログラミング教育の必要性を訴えかける動画を発表している。

日本でも経済産業省が発表した、IT人材の最新動向と将来推計に関する調査結果[3]によると、2020年に36.9万人、2030年には78.9万人のIT人材が不足すると予測されており、文部科学省が2020年度からを目標に小学校におけるプログラミング教育を必修化すると発表している。他にも2011年より、ライフイズテック株式会社が中学生・高校生のためのIT教育プログラム「Life is Tech!」[4]をスタートし、現在では15000人以上の中学生・高校生が参加しており、国内最大級のプログラムとなっている。また、経済産業省が主催しているU-22プログラミングコンテスト[5]も今年度から小学生部門が新設された。

現在、プログラミング教育ではビジュアルベースなプログラミング言語、特にScratch[6]が利用されている。Özgenの報告[7]では、Scratchが他のプログラミング教材よりもプログラミングスキルや論理的思考力向上の面できわめて

<sup>1</sup> 中京大学 工学部情報工学科

<sup>2</sup> 中京大学 情報科学研究科

高い効果が得られている。しかしながら、Scratch によるプログラミングそのものに対する意識についてアンケートを行っている [8] では、「親しみやすく抵抗感も感じない」「社会で役立つものである」という点に関しては意識が高いのに対し、「プログラミングに自ら積極的に関わっていききたいかどうか」という点ではやや低い意識であるという結果が得られている。

そこで、我々はビジュアルベースなプログラミングに別の要素を取り入れることで「プログラミングに自ら積極的に関わっていききたいかどうか」という点でも意識が高くなるようなシステムを開発できないかと考えた。取り入れる要素として我々は現実世界での「ものづくり」に注目した。初等教育では「図画工作」、中等教育であれば「技術・家庭」において「ものづくり」を行う機会があり、児童たちに馴染み深いからである。

我々は現実世界での「ものづくり」をプログラミングに取り入れるために 3D スキャナを利用した。3D スキャナとは、スキャンする物体の表面の凹凸を感知し、 $x, y, z$  の座標データをもった無数の点を取得し、そのデータから 3D モデルを作り出すことが出来る機器である。以前は [9] にあるように工業デザインや医療といった業務に利用されるものであったが、2017 年現在では [10] のように個人用の 3D スキャナも 3 万円ほどで販売されており一般家庭にも普及してきている。

我々はこの 3D スキャナを用いて現実世界の「ものづくり」要素を取り入れた児童が関心を持って楽しく取り組めるプログラミング教育システムを開発した。

## 2. 関連研究

近年では児童に対するプログラミング教育は大いに注目を集めており、様々な研究が行われている。ここでは児童に対するプログラミング教育を目的とした教材の事例を示し、本研究との位置づけと期待される効果を示す。

### 2.1 ビジュアルプログラミング

今日、初等教育向けのビジュアルプログラミング言語は多く開発されている。その中の一つに MIT Media Laboratory が開発した Scratch がある。これはパズルを組み立てるように、ブロックとして用意されている「命令」を組み合わせるプログラミング言語である。簡単なマウス操作だけでプログラミングができるため、プログラミング入門教材として広く使用されている。森らは Scratch を用いて小学 4 年生向けにプログラミングの授業をデザインし実践した [11]。26 時間の授業を通し小学校段階でプログラミングの学習が可能であることを確認した。また、Kono らは AT と呼ばれるビジュアルプログラミング環境を開発した [12]。AT は直感的な操作を重視しており、PlanBlock と呼ばれるプログラミングの構文を図表化したものをつなげること

でプログラミングを行う。Kono らは大学のクラスで AT を導入し、学生から高評価を得た。従来のビジュアルプログラミングでは、すべての作業が仮想空間内で完結しているが、提案システムでは児童になじみ深い現実空間でのものづくり要素を取り入れているため、より児童の興味を引けると考える。

### 2.2 ロボットプログラミング

教育用のロボットとして LEGOMINDSTORM [13] がある。LEGOMINDSTORM は複数のモーター、センサ、レゴブロックを組み合わせてロボットを制作し、専用のソフトウェアでそのロボットの動作をプログラミングすることができる。土居らは大学 1 年生を対象に LEGOMINDSTORM を用いたロボットの製作体験型授業を実施した [14]。授業に対するアンケートを全学年で実施し、満足度、学習意欲の向上を確認した。また、Michael らは低価格な教育用ロボット AERobot の開発を行った [15]。AERobot は一台 10 ドル以下で構築することができ、ビジュアルプログラミング環境により、プログラミング経験のないユーザでもロボットを制御することができる。上記の研究や従来の教育用ロボットでは、専用のブロックを使用するため造形に制限がある。しかし、提案システムでは 3D スキャナを使用し、あらゆるものをスキャンし取り込むことができるため、ユーザはより自由な創造が行える。

### 2.3 プログラミング教育を目的としたシリアスゲーム

Ioannis らは早期プログラミング教育を目的とした 3D ビデオゲーム、PlayLOGO3D の開発を行った [16]。PlayLOGO3D は、簡単なコマンドで三次元空間内のオブジェクトを操作し、オブジェクトの軌跡で図形を描写する。ユーザはゲームでの体験を通し、プログラミングの基礎知識だけでなく空間把握能力も身に着けることができる。

### 2.4 3D スキャナを用いた教育

3D スキャナは教育現場でも利用されている。太田は中学校で 3D スキャナを取り入れた理科授業を行った [17]。3D スキャナで頭蓋標本をスキャンし 3D プリントすることで標本のレプリカを複数制作した。これにより、標本をもとにした具体的な比較や考察を伴う学習活動を導入することができた。

## 3. 提案システム

3D スキャナを用いることで現実世界での「ものづくり」を取り入れ、ユーザのプログラミングに対する興味を引くプログラミング教育システムについて提案する。

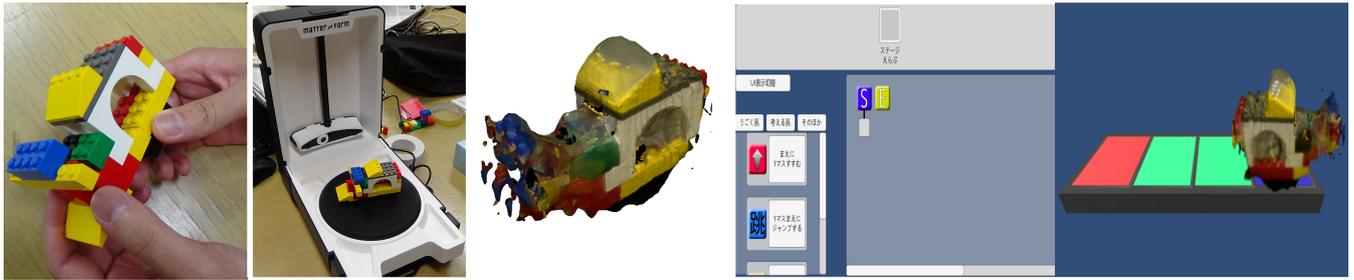


図 1 システム全体図:左から立体物作成、スキャンしたデータ、OBJ 形式に変換したファイル、プログラミング画面、迷路脱出の課題実行画面

### 3.1 対象

提案システムでは初等教育から中等教育の児童を主対象とし、加えてプログラミングを学ぼうとしている人や論理的思考を身に着けたいと考えている人が利用することも想定している。

### 3.2 システム構成

システムの全体図を図 1 に示す。ユーザは始めに現実空間で、システムで使用する任意の立体物を LEGO 等で作成する。次に作成した立体物を 3D スキャナでスキャンし OBJ 形式のファイルを作成する。OBJ 形式とは物質の形状、配置などを記憶したファイルフォーマットである。OBJ ファイルを作成した後システムに 3D モデルとして取り込む。そしてユーザは我々が開発したビジュアルベースのプログラミング言語を用いて 3D モデルを制御し、我々があらかじめ用意した課題に取り組む。

課題は自由な発想が発揮できる迷路脱出問題を用意した。迷路脱出問題とは、迷路内に配置された 3D モデルを制御しスタートからゴールへ移動させるようにプログラミングを行う問題である。迷路内にはオブジェクトの造形次第では通ることのできない障害物が用意されており、これにより立体物を作成する際に工夫の余地が生まれる。この問題では、ユーザの成績を付けるためにスタートからゴールへ移動するまでのユーザのプログラムにおける行動回数をカウントしており、あらかじめ我々が設定した目標行動回数と比較して成績が付けられるようになっている。

### 3.3 迷路脱出問題

迷路脱出問題は、迷路内に配置された 3D モデルを、赤色のマスから青色のマスまで移動することが目的である。あらかじめ用意された 3D モデルを使用して動作の指定方法の学習を目的とする「チュートリアルステージ」と、現実世界で制作した 3D モデルで課題に挑戦する「本番ステージ」の 2 種類が存在する。チュートリアルステージの例を図 2 に、本番ステージの例を図 3 に示す。

#### 3.3.1 チュートリアルステージ

チュートリアルステージでは動作の指定方法の学習を目

的としている。図 2 のようにマス目で区切られたステージでは例えば「前に進む」を指定すれば「1 マス前に動く」のように動作の組み立てが容易であり、動作を指定する方法を学習することができる。

#### 3.3.2 本番ステージ

本番ステージでは現実世界で制作した 3D モデルの形に合わせてオブジェクト同士の衝突判定を行っており造形の影響が出るようなステージとなっている。図 3 では、一見すると直進し続ければゴールできるように見えるが、途中で突起状の障害物が存在しており 3D モデルの造形によってはそこで転倒して動けなくなっている。図 4 のような背が高く底面が狭い 3D モデルでは突起状の障害物を通過する際にバランスを崩し、転倒し動けなくなってしまう。一方で図 5 のような背が低く底面が広い形の 3D モデルであれば突起状の障害物を通過しても転倒することはない。動けなくなった場合には「別のルートを通ってゴールを目指すように動作を指定する」か「どういった造形であれば障害物を突破できるかを考えて新しく 3D モデルを現実世界で制作する」といった手法でゴールを目指す。

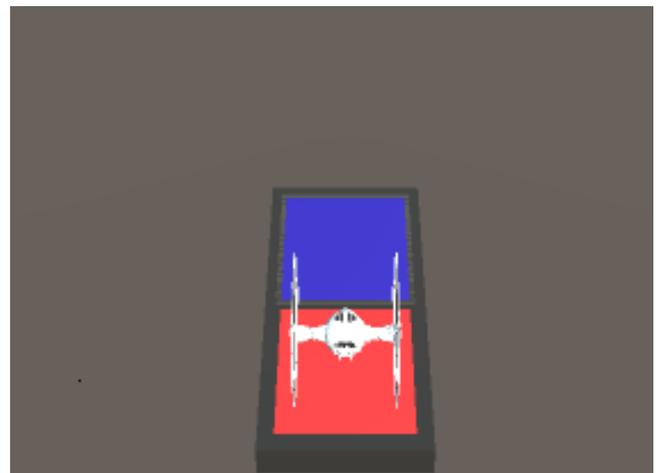


図 2 チュートリアルステージ例

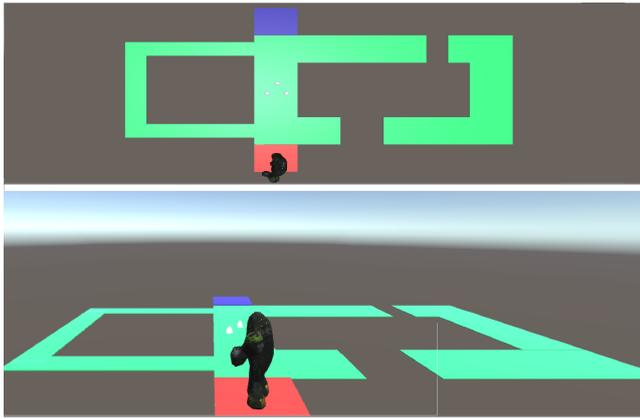


図 3 本番ステージ例 上図はステージを真上から、  
下図は斜めから見たもの

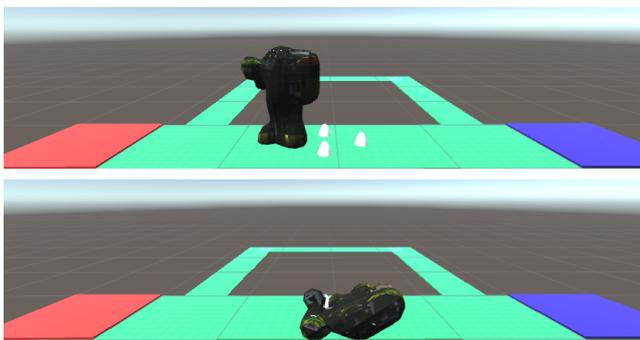


図 4 背の高いオブジェクトを使用した場合

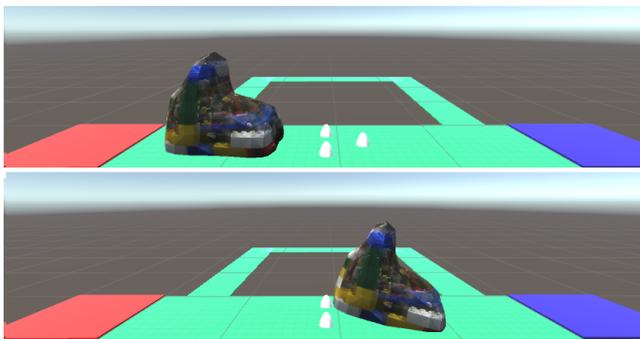


図 5 背の低いオブジェクトを使用した場合

### 3.4 プログラミングエディタ部

提案システムではマウス操作でプログラミングを行える、専用のエディタを実装した。

#### 3.4.1 エディタ概要

エディタ画面を図 6 に示す。提案システムにおけるプログラミングとは、3D モデルの動作を定義してある「命令ブロック」をフローチャート図のようにつなぎ合わせ、3D モデルの動作をプログラムすることを指す。ユーザはエディタ上で、命令ブロックの追加、連結、削除を繰り返しながらプログラミングを行う。

#### 3.4.2 実装されている命令ブロック

提案システムのプログラミングに用いることのできる命令ブロックを、表 1 に示す。命令ブロックは、命令の内容

の特性によって分類されており、モデルの移動や回転に関わる命令ブロックは「うごく系」、条件分岐に関わる命令ブロックは「考える系」、ループ端といったような「うごく系」や「考える系」に分類されないその他の命令は「そのほか」の 3 種類に分類されている。この分類は、後述する命令ブロック一覧の表示の際に利用される。

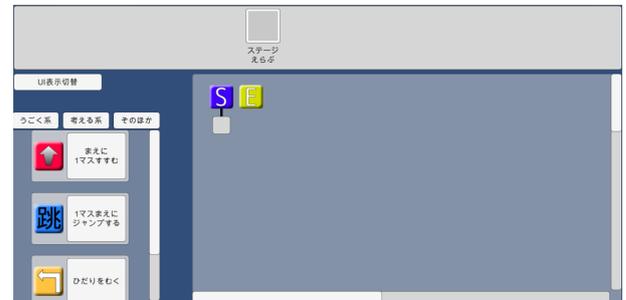


図 6 エディタ画面全体図

### 3.5 プログラミングエディタの操作方法

具体的なプログラミングエディタにおける操作を以下に示す。

#### 3.5.1 命令ブロックの追加

命令ブロックの追加操作は、画面左部に表示されている命令ブロック一覧の中から、追加したい命令ブロック上でドラッグ操作を開始し、画面中央から右部の編集領域の空欄部分にドロップすることで命令ブロックを追加することができる。編集領域の表示部分を図 7 に示す。

命令ブロック一覧の表示部分を、図 8 に示す。命令ブロック一覧は、命令ブロックの分類によって表示内容がフィルタリングされている。起動直後の状態では「うごく系」の命令ブロックのみが一覧表示されている。フィルタリングの切り替えについては、一覧の上部に表示されている「うごく系」「考える系」「そのほか」の各分類の名前が表記されたボタンを押すことで行う。



図 7 編集領域

表 1 使用可能な命令ブロック一覧

画像	命令名	分類	命令の概要
	前に進む	うごく系	前方に 1 マス分移動
	後ろに進む	うごく系	後方に 1 マス分移動
	1 マス飛び越える	うごく系	跳躍動作を行い、2 マス先に跳びながら移動
	左を向く	うごく系	左に 90 度回転
	右を向く	うごく系	右に 90 度回転
	前方に壁があるか	考える系	前方に壁があるかを判定、ある場合は右の接続端、ない場合は下の接続端の処理へ移行
	ループ始端	そのほか	ループ処理を開始以降、ループ終端までの処理を繰り返す
	ループ終端	そのほか	ループ処理の終端を表現 ループ始端からこの命令の間の処理を繰り返す



図 8 命令ブロック一覧

### 3.5.2 命令ブロックの連結と連結の解除

すでに置かれている命令ブロックについて、次の命令ブロックが設定可能なものであれば、図 9 のように、命令ブロックに灰色の矩形の接続端が付属している。

すでに追加された命令ブロックをドラッグ操作し、接続端にドロップすることで、その次の命令ブロックを設定できる。また、接続端をクリックし、接続したい命令ブロックをクリックすることで、命令ブロック同士が線で接続され、次の命令ブロックを設定することができる。連結された状態の命令ブロックは、ドラッグ操作で接続端から離すことで、連結を解除することができる。

### 3.5.3 命令ブロックの削除

削除したい命令ブロックを編集領域の領域外にドラッグアンドドロップすることで、編集領域上に配置されている



図 9 命令の接続端

命令ブロックを削除することができる。

### 3.5.4 プログラムの実行

画面上部のツールバー上に存在する「ステージ選ぶ」表記のあるボタンを押下することで、「ステージ選択ダイアログ」が表示される。プログラムを実行する「ステージ」をプルダウンメニューから選択し、「OK」ボタンを押すことで、エディタの背景で、プログラムが実行される。

### 3.5.5 プログラム例

以上を踏まえ、提案システムでの具体的なプログラミングの流れを以下の図 10 に示す。図 10 は、渦巻状の道で、3D モデルをスタートからゴールへ移動させるプログラムである。

## 3.6 3D スキャナ

提案システムでは 3D スキャナを使用するにあたり作成物のサイズとスキャン精度を考慮し Matter and Form[18]を使用した。立体物を作成するにあたり作成物のスキャンにかかる時間が高さ 5cm のものでおよそ 7 分、10cm のもので 14 分かかり、大きさが極端なものは作成に時間がかかるということと、Matterandform のスキャン可能な幅が 18cm である事から 5~15cm 四方の四角柱に収まるものをユーザは作成する。

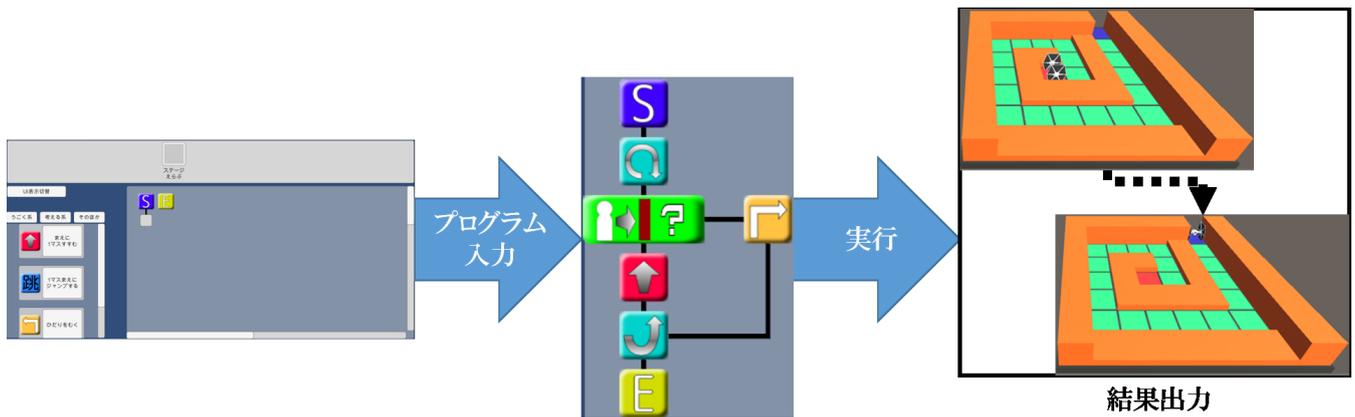


図 10 エディタによるプログラミングの流れ：ユーザはエディタ画面でプログラムを作成し、プログラムをステージ上の 3D モデルに適用し、実行結果を確認する。

## 4. 評価

### 4.1 実験内容

被験者 8 人 (中学生 2 人、大学生 6 人) に対して、提案システムがユーザに「プログラミングに自ら積極的に関わっていききたい」と思わせ、興味を引けるシステムであるかの検証を目的とする評価実験を行った。評価実験では提案システムと比較対象として Scratch を体験し、アンケートに回答してもらった。実験手順は以下の通りである。

- (1) レゴブロックによる 3D モデルの制作
- (2) 提案システムを体験
- (3) Scratch を体験
- (4) 提案システムに関するアンケート
- (5) Scratch に関するアンケート

提案システムを体験している図 11 に示す。



図 11 提案システム体験風景

#### 4.1.1 提案システムの体験

提案システムでは、ユーザは 3D モデルの動作を指定して迷路脱出を目的とするステージを体験した。ステージはあらかじめ用意された 3D モデルで挑戦するチュートリアルステージと現実世界で制作した 3D モデルで挑戦する本番ステージを合わせて 12 ステージ用意した。

#### 4.1.2 Scratch の体験

Scratch の体験では初めに Scratch の基礎であるスプラ

イト (二次元画像) の動かし方を学習してもらった。学習内容は以下の通りである。

- (1) Scratch の基本操作
- (2) スプライトの x 軸方向の移動方法
- (3) ループの使用方法

学習後は、学習成果を確認する目的であるチャレンジ課題に挑戦してもらった。チャレンジ課題ではスプライトが画面の端を往復するシーンを制作してもらった。図 12 に被験者が制作したコードの一例を示す。

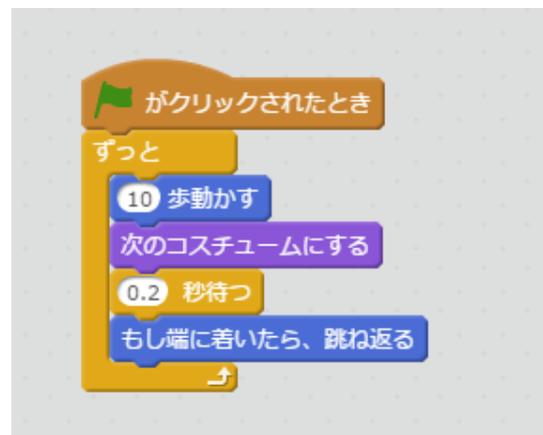


図 12 被験者が制作したコード

#### 4.1.3 アンケート

アンケート項目には ARCS モデル [19] を採用した。ARCS モデルとは John M. Keller が提唱した学習者の学習意欲に働きかけ、「学びたい」と思わせるための方策を考えるための形式である。学習意欲の問題と対策を、注意 (Attention) ・ 関連性 (Relevance) ・ 自信 (Confidence) ・ 満足感 (Satisfaction) の 4 つに分け、さらにそれぞれの要因を 3 つの小要因に分けた計 12 の要因と、それぞれの要因に対応した動機的设计の手順を提案したものである。今回この ARCS モデルを用いることで児童の学習意欲を測定した。

実際に被験者に行ったアンケート内容と対応する ARCS モデルの要因をまとめたものが表 2 である。

表 2 アンケート内容

大要因	小要因	アンケート内容
注意	知覚的喚起	やってみたいと思えたか
	探究心の喚起	こうしたらできそうなど、工夫の余地があったか
	変化性	続けてやっていきたいと思えたか
関連性	親しみやすさ	システムに抵抗を感じたか
	目的指向性	プログラミングができるようになりそうか
	動機との一致	プログラミングを学ぶ上でこの教材は意味があると感じたか
自信	学習欲求	学習の目的を理解できたか
	成功の機会	問題を回答できたか、もしくはできそうだったか
	コントロールの個人化	自身の力で問題を解決できたと思えましたか
満足感	内発的な強化	プログラミングが以前よりできるようになったと感じたか
	外発的報酬	ステージをクリアした際に満足感を得られましたか
	公平さ	不満な点はないか

## 4.2 評価結果

### 4.2.1 中学生に対する評価

提案システムは初等教育から中等教育の児童への利用を想定しているため 12~13 歳の中学生 2 人に提案システムを体験してもらった。児童 2 人のアンケート結果を図 13 に示す。アンケート結果より、提案システムの Attention のスコアが比較対象である Scratch のスコアを上回ったことがわかる。このことより、現実世界での「ものづくり」を取り入れたことにより被験者の興味を引き出すことができたと考えられる。また、Confidence と Satisfaction の項目も Scratch を上回ったことがわかる。これは提案システムの課題の目的が迷路をゴールするという明確なものであったためであると考えられる。

### 4.2.2 大学生に対する評価

提案システムの主対象は児童であるが、プログラミングを学ぼうとしている人や論理的思考を身に着きたい人が利用することも想定している。そのため、大学生 6 人に提案システムを体験してもらいアンケートに回答してもらった。大学生 6 人のアンケート結果を図 14 に示す。アンケート結果より、提案システムの Attention のスコアが比較対象である Scratch のスコアを上回ったことがわかる。このことより、大学生であっても提案システムが被験者の興味を引き出すことができたと考える。また、Scratch の Confidence のスコアが児童と比較して高い点は、Scratch の課題が大学生にとっては容易であったためだと考えられる。

### 4.2.3 統計的分析

今回のアンケート結果より、提案システムと Scratch の ARCS モデルにおける各項目のスコアの差が有意なものであったかを分析した。まず、被験者 8 人の提案システムと Scratch のアンケートからそれぞれ各項目のスコア平均を算出した (有効数字 3 桁)。その後それらで一標本  $t$  検定を有意水準 5% で行ったところ、以下の表 3 ような結果が得られた。

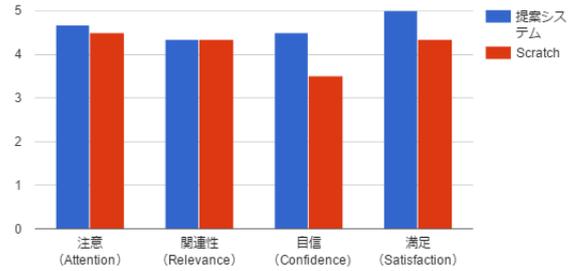


図 13 ARCS モデルによるアンケート結果 (児童 2 人)

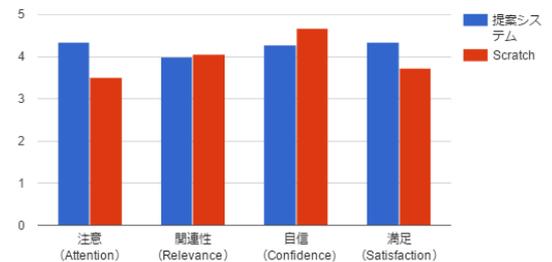


図 14 ARCS モデルによるアンケート結果 (大学生 6 人)

表 3 一標本  $t$  検定の結果

	Attention の平均スコア								$p$ 値
提案システム	5.00	5.00	4.00	4.67	4.33	3.00	4.67	4.67	0.0520
Scratch	2.33	3.67	3.00	4.67	4.67	2.67	5.00	4.00	
	Relevance の平均スコア								$p$ 値
提案システム	4.33	3.67	4.33	3.33	4.33	3.33	3.33	2.67	0.3313
Scratch	4.00	4.00	3.00	3.67	3.67	2.67	3.67	3.67	
	Confidence の平均スコア								$p$ 値
提案システム	4.00	5.00	3.67	4.67	3.33	5.00	4.33	4.67	0.5495
Scratch	4.00	5.00	4.67	4.67	5.00	4.67	3.00	4.00	
	Satisfaction の平均スコア								$p$ 値
提案システム	5.00	4.33	3.67	5.00	4.33	3.67	5.00	5.00	0.0054
Scratch	3.67	4.33	3.33	3.67	4.33	3.00	4.33	4.33	

表より、Satisfaction の項目では  $p < 0.05$  であるため有意差が得られた。一方で Attention, Relevance, Confidence の項目では  $p > 0.05$  であるため有意差は得られなかった。

### 4.2.4 考察

統計的分析の結果、Satisfaction の項目では有意差を得ることができた。これは提案システムの課題が迷路を脱出するという明確なものであったため被験者が課題を達成した際に大きな満足感を得ることができたためだと考えられる。また、有意差は得られなかったが中学生・大学生の両方で提案システムの Attention のスコアが Scratch を上回ることができた。これは提案システムが、現実空間での「ものづくり」を取り入れた機能を有しているためだと考えられる。Scratch との機能の差異を表 4 に示す。提案システムでは現実空間で制作した立体物を 3D スキャンし PC 内に取り入れ、三次元空間内を移動する。被験者からも「自身で制作した立体物を動かして楽しかった」という意見があった。これらの要因が提案システムの Attention

のスコアを高めたと考える。一方で用意されている命令の数は Scratch が提案システムより多く、より細かく動作をプログラミングすることができる。そのため、Scratch の Relevance のスコアが提案システムと比較して高かったと考える。

表 4 Scratch と提案システムの比較

	Scratch	提案システム
操作対象	スプライト (二次元画像)	3D モデル
移動空間	二次元空間	三次元空間
動作を指定する命令の数	17	8

## 5. おわりに

今回、我々は 3D スキャナを用いて現実世界の「ものづくり」要素を取り入れたプログラミング教育システムを開発した。

評価実験の結果、ユーザが「プログラミングに自ら積極的に関わっていきたい」と思い、興味を持つかという点において提案システムは従来のプログラミング教育に使用されるシステムよりも優れているように見えた。しかし、統計学的に有意な差は得られなかった。

今後の展望として被験者から要望があった、「作成した 3D モデルとプログラムを使い、3D モデル同士を戦わせる」課題を実装することを検討している。現状では「迷路をゴールする」課題のみが実装されており、プログラミング等の作業がユーザー一人で完結するようになっているが、これによりユーザ間での競争性を持たせる事ができると考える。そうすることで、ユーザは他のユーザのプログラムを参考にしてより良いプログラムを作ろうとする、といったような心理が働くようになり、ユーザ同士でプログラミングスキルを磨きあう状態が発生することが期待できる。

加えて、現段階では出力結果を見るための視点が固定されているため、カメラを自由に操作する機能を追加したり、出力結果を AR 表示する機能を追加する。これらの機能により、ユーザは「迷路をゴールする」課題であれば迷路の構造を、「作成した 3D モデルを戦わせる」課題であれば戦闘中の様子を自由な角度から見るができるようになる。様々な角度から見ることで、固定視点では気づき辛かった部分に注目することができ、異なる視点に立ったプログラムを作成することができるようになることが期待できる。

## 参考文献

[1] 文部科学省:「諸外国におけるプログラミング教育に関する調査研究」報告書, 入手先 ([http://jouhouka.mext.go.jp/school/pdf/programming\\_syogaikoku\\_houkokusyo.pdf](http://jouhouka.mext.go.jp/school/pdf/programming_syogaikoku_houkokusyo.pdf))

[2] code.org 公式サイト入手先 (<https://code.org/>)

[3] 経済産業省:「IT 人材の最新動向と将来推計に関する調査結果」, 入手先 (<http://www.meti.go.jp/>)

press/2016/06/20160610002/20160610002.pdf)

[4] Life is Tech!公式サイト, 入手先 (<https://life-is-tech.com>)

[5] U-22 プログラミングコンテスト 2017 公式サイト, 入手先 (<http://www.u22procon.com/>)

[6] scratch 公式サイト, 入手先 (<https://scratch.mit.edu/>)

[7] Özgen KORKMAZ: The Effect of Scratch- and Lego Mindstorms Ev3-Based Programming Activities on Academic Achievement, Problem-Solving Skills and Logical Mathematical Thinking Skills of Students, Amasya University, Malaysian Online Journal of Educational Sciences (2016) 入手先 (<http://files.eric.ed.gov/fulltext/EJ1106444.pdf>)

[8] 山田耕太郎: Scratch によるプログラミング教育の実践と評価, 比治山大学紀要, 第 22 号 (2015)

[9] ARTEC 3D: 3D スキャナ用途, 入手先 (<https://www.artec3d.com/ja/applications>)

[10] XYZ プリンティングジャパン: XYZprinting ハンドヘルド 3D スキャナー 1.0A, 入手先 (<http://jp.store.xyzprinting.com/jp-ja/catalog/scanner/HHS10A?.ga=2.216006865.654697654.1493871884-1039428999.1493871884>)

[11] 森秀樹, 杉澤学, 張海, 前迫孝憲: Scratch を用いた小学校プログラミング授業の実践: 小学生を対象としたプログラミング教育の再考, 教育実践研究論文 (2011)

[12] Nao Kono, Hisayoshi Kumimune, Tatsuki Yamamoto, Masaaki Niimura : Development and Evaluation of Functions for Elementary/Secondary Programming Education: The Visual Programming Environment "AT", International Journal of e-Education, e-Business, e-Management and e-Learning, vol.7, no.1, pp.13-23, March 2017.

[13] LEGOMINDSTORM 公式, 入手先 (<https://www.lego.com/ja-jp/mindstorms>)

[14] 土居隆宏, 平澤一樹, 河合宏之竹井義法, 鈴木亮一, 小暮潔, 出村公成: 初年次教育における LEGO Mindstorms を用いたロボット製作体験の試み, 工学教育研究;KIT progress(2016)

[15] Michael Rubenstein, Bo Cimino, Radhika Nagpal, Justin Werfel: AERobot: An affordable one-robot-per-student system for early robotics education, ICRA(2015)

[16] Ioannis Paliokas, Christos Arapidis, Michail Mpimpitso: PlayLOGO 3D: A 3D Interactive Video Game for Early Programming Education: Let LOGO Be a Game, VS-GAMES(2011)

[17] 太田聡: 科学的思考力・判断力・表現力を高める理科学習の展開: 細胞の営みを実感し, 未解決課題への科学的な思考を育成する教材開発, 滋賀大学教育学部附属中学校研究紀要, 第 56 集 (2014) 入手先 (<http://hdl.handle.net/10441/13186>)

[18] Matterandform HP, 入手先 (<https://matterandform.net/>)

[19] Keller, John M: Motivational Design for Learning and Performance The ARCS Model Approach