

AR を用いたロボットプログラミング学習支援の提案

塩澤秀和^{1,a)} 松本祐衣^{1,b)}

概要: 子ども向けのプログラミング教育では、プログラムを小型のロボットに転送して実際に動かすロボットプログラミング教材が人気である。これは、目の前でロボットが動くことで初心者にもプログラムの動きが視覚的にわかりやすく、複数の子どもによる協同学習もしやすいのが利点である。しかし、ロボットを用いたプログラミング学習でも、学習者にとってプログラムが少し複雑になると、ロボットの動作と実行中のプログラムの該当コードとの対応の理解が、容易とはいえなくなる。そこで、本研究では、プログラムの実行中の命令や変数の値など、ロボットの内部状態や実行手順の経過を AR (拡張現実感) 技術を用いてロボットの映像に重ねてリアルタイムに可視化し、より子どもの理解を促すロボットプログラミング教材の試作を行った。プログラミング言語としては、物理的なカードを並べることによってプログラミングを行うタンジブルプログラミングを採用した。本システムの特徴のひとつは、実行中のプログラムの情報がロボットから PC に送信され、プログラムの実行手順に合わせてリアルタイムに AR 表示が更新されるという、双方向の情報通信を利用した AR システムとなっていることである。

A Proposal to Use AR to Support Robot Programming Learning

HIDEKAZU SHIOZAWA^{1,a)} YUI MATSUMOTO^{1,b)}

1. はじめに

近年、子どもを対象とした早期のプログラミング教育が世界的に注目を集めており、義務教育のカリキュラムに導入する国も増えている。例えば、イギリス (イングランド) では、2014 年から小学校でコンピュータサイエンスを扱う「コンピューティング」が必修科目になり[1]、単なるコンピュータ機器の操作だけではなく、プログラミングに関する内容を教えている。

日本でも 2020 年度から実施される学習指導要領では、初等教育におけるプログラミング教育の必修化が盛り込まれた。これは、プログラミングを通して論理的な思考力である「プログラミング的思考」[2]の力を養うことを目的としており、プログラミング言語の文法を習得させてソフトウェア開発技術者を育成することを直接的に目的とするものではないとされる。

このような背景のもと、最近では子どもを対象としたプログラミングの教育方法や学習教材が数多く提案・市販されている。その中でも、子供向けのプログラミング教室やワークショップで人気なのが、小型のロボットを用いたプログラミング学習教材である。これは、学習者が PC 画面やタブレット上でプログラムを組み、それをロボットに搭載したマイコンに無線または有線で転送し、ロボットがそ

のプログラムに従って自立して動作するものである。

小型のロボットを用いたプログラミング学習の利点は、ロボットが動くことで初心者にもプログラムの動きが視覚的に理解しやすく、また複数の学習者がプログラムの動作結果を目の前で共有できるので、子どもたちの協同学習にも向いていることである。

しかし、小型のロボットを用いたプログラミング学習であっても、ロボットに転送された後のプログラムは学習者の手を離れて自立して動作するので、プログラムの複雑度が増すにつれて、学習者にとって実行中のプログラムと目の前のロボットの動作との対応関係の把握は容易ではなくなる。なぜなら、プログラムで実行中の命令や変数の値など、ロボットの内部状態や実行手順の途中経過は、学習者にとって直接は見るできないからである。

そこで、本研究では、小型のロボットを用いたプログラミング学習において、実行中のプログラムの情報を AR (Augmented Reality ; 拡張現実感) 技術を用いてロボットの映像にリアルタイムに重ねて可視化し、学習者である子どもの理解を深める教材を開発した。これは、ソフトウェア開発におけるデバッグの機能を AR によって現実空間に融合させる試みとも位置付けられる。

また、プログラミング言語としては、物理的なブロックやカードを配置することによってプログラミングを行うタンジブルプログラミングと呼ばれる方法を採用した。これによって、プログラミング初心者の子どもにも扱いやすく、協同学習に適した教材システムの実現を目指した。

1 玉川大学工学部ソフトウェアサイエンス学科
Department of Software Science, Tamagawa University
a) shiozawa@eng.tamagawa.ac.jp
b) 2019 年 3 月卒業

2. 関連研究

子ども向けのプログラミング言語としては、Scratch [3] または、それに類似したブロック型のビジュアル言語が採用されることが多い。これは、分岐、反復、表示、演算などの基本的な要素を、それぞれ特徴的な形のブロックで表現するものである。プログラミング初級者は、細かい文法の理解や活用に苦しむことなく、ブロックを組み立てるようにプログラムを作成することができる。

文献[4]では、Scratch のようなブロック型言語と Java のようなテキスト型言語を組み合わせた講義を行った調査が紹介されている。この講義を受けた学生は、従来のテキスト型言語だけの講義を受けた学生に対して、Java 言語の最終試験で平均 10 パーセント以上成績を向上させたことが報告されている。

このブロック型のビジュアル言語を、コンピュータの画面上ではなく、物理的なブロックやカードによって実装したものが、タンジブルプログラミングである。その初期の例として、アルゴブロック [5] がある。このシステムでは、電子回路が内蔵された立方体のブロックに「前に進む」「くりかえす」など、その機能を表すマークが描かれており、ユーザは、これらのブロックを端子で結合していくことでプログラミングを行う。この研究では、ブロックという実体があることで、子どもにとってプログラムの構造がより理解しやすく、子ども同士の協同学習にもふさわしいことが報告されている。

また、最近、子供向けのプログラミング教室やワークショップで普及しているのが、小型のロボットを用いたプログラミング学習教材 (図 1) [6][7][8] である。これは、学習者が PC 画面やタブレット上でプログラムを組むと、ロボットがそのプログラムに従って自立して動作するものである。この方式の利点は、ロボットが動くことで初心者にもプログラムの動きが視覚的に理解しやすく、また複数の学習者がプログラムの動作結果を目の前で共有できるので、子どもたちの協同学習にも向いていることである。

さらに、タンジブルプログラミングとロボットプログラミングを組み合わせたプログラミング学習システム [9] も提案されている。この研究では、タンジブルプログラミングを実装するために、AR マーカーが印刷されたカードを用いている。学習者はコードを表すカードを適切な順序で並べてプログラミングを行う。システムはそれをカメラで撮影し、AR マーカーを認識してプログラムを読み取り、Bluetooth 通信でロボットにプログラムを転送する。

その他にも、小さなブロックを順番に操作盤にはめ込むことでロボットプログラミングができる Cubetto [10] という製品もある。これは、PC などを介さず、操作盤とロボットが直接つながっているため、PC が扱えない幼児でもプログラミングをすることができる。

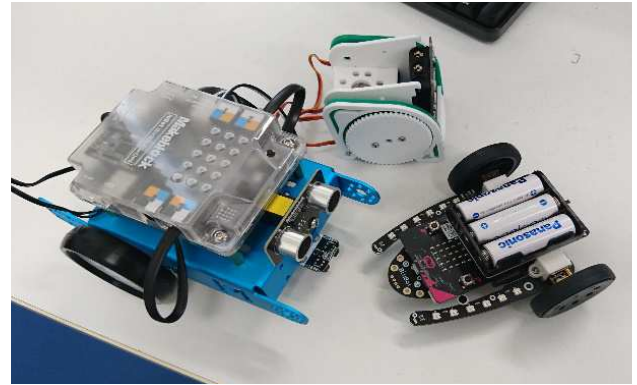


図 1 ロボットによるプログラミング学習教材

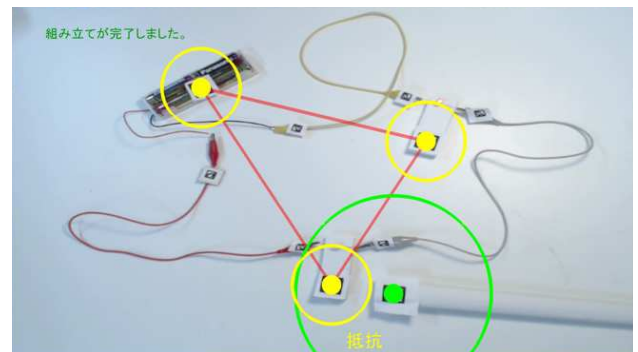


図 2 回路実験における接続の AR 可視化

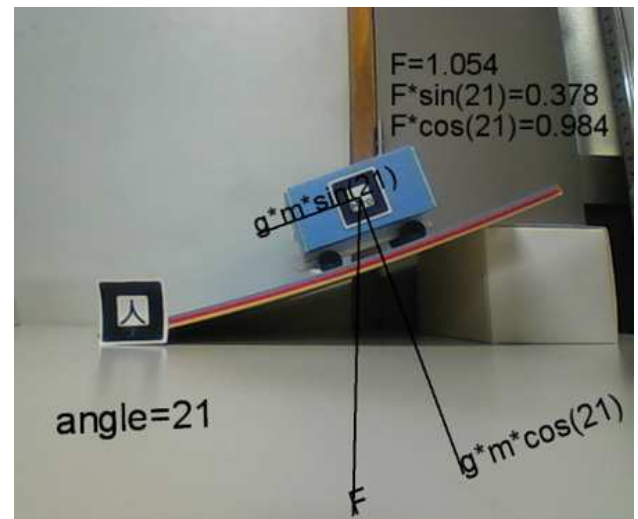


図 3 力学実験における力の AR 可視化

学習教材に AR を利用する試みとしては、著者らもマーカー型 AR とセンサーを用いた物理実験教材 (図 2, 図 3) の開発を進めている [11][12][13]。これは、実験で用いる測定装置、その他の器具、環境などの教材に取り付けたセンサーから、PC に測定データを取り込み、そのデータをグラフなどで分かりやすく可視化して、カメラで撮影した教材の映像に画面上で合成するものである。

AR を利用した既存の多くの教育システムは、AR を主に

コンピュータで生成した情報の提示に用いるが、我々は、現実空間の教材にセンサーを組み込み、そこから得たリアルタイムな実測情報を AR で提示するという双方向的な手法を提案している。図 2 は回路の素子の接続状況を、図 3 は台車にかかる力のベクトル（実測値）を、実写映像に重ねて CG でリアルタイムに可視化したものである。

3. 本研究の提案

本研究では、小型のロボットを用いたプログラミング学習において、タンジブルプログラミングでプログラミングを行うことができ、AR（拡張現実感）技術を用いて実行中のプログラムの情報をロボットの映像にリアルタイムに重ねて可視化する教材を提案する。AR 技術を応用することで、学習者にとって、プログラムで実行中の命令や変数の値などロボットの内部状態や実行手順の途中経過を視覚的に把握しやすくなり、プログラミングへの理解度が向上すると考えている。

学習者は、まず図 4 に示すように、物理的なカードを用いてプログラミングを行う。システムは、各カードに付与された AR マーカーをカメラで撮影し、それを認識することでコードの配置を認識し、プログラムとして読み込む。そして、読み込んだプログラムに適切な処理を施した後、ロボットに送信することでプログラムを実行する。

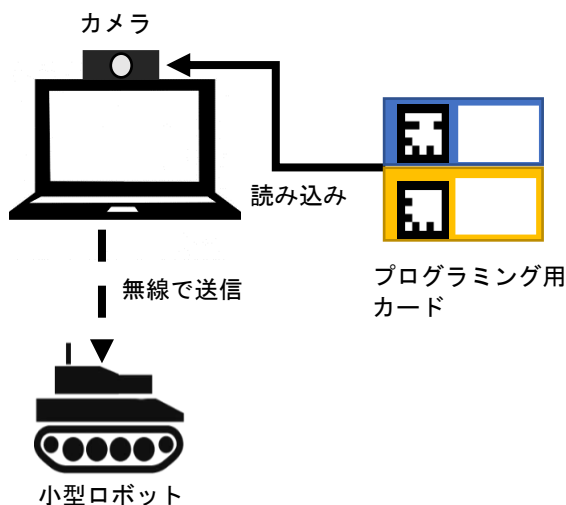


図 4 ロボットのタンジブルプログラミング

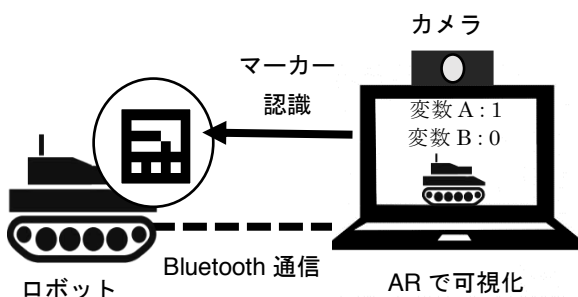


図 5 AR による内部データの可視化

図 5 に示すようにロボットには AR マーカーが付与されており、学習者はそれを PC やタブレット端末を通して AR によって見ることで、変数や繰り返しなどのプログラムの情報をロボットに重ねてみるができる。ロボットは、本機能を実現するために、PC と無線でリアルタイムに情報を送受信できるものを利用する。

本研究で重要な点は、実行中のプログラムの情報がロボットから送信され、プログラムの実行手順に合わせてリアルタイムに更新されるという、双方向の通信を利用した AR システムとなっていることである。

4. システムの開発

4.1 使用したロボットプログラミング教材

本研究では、Makeblock 社が市販しているプログラミング教育向けロボットの mBot [6] を使用した。mBot は、Arduino [14] の技術に基づいて開発されているため、多様なセンサーが使用可能であり、Arduino IDE でプログラミングおよびロボットへのプログラム送信が可能である。

本システムでは、この mBot に AR 表示を行う PC との無線通信を可能にする Bluetooth モジュールをつけ、さらに AR による付加情報の表示を実現するために、図 6 の写真のように背面に AR マーカー (NyARToolKit [15] の NyID マーカー) を付与した。

4.2 タンジブルプログラミングの実現

本システムでは、学習者は机の上で「前進」や「停止」などの処理を表す物理的なカードを並べることでプログラミングを行う。カードは、表 1 に示した約 40 種類が用意されており、Scratch を参考にして学習者に分かりやすいように機能ごとに色分けされている。

各カードにはシステムで認識するためのマーカー (AR マーカー) が印刷されている。本システムではマーカーの

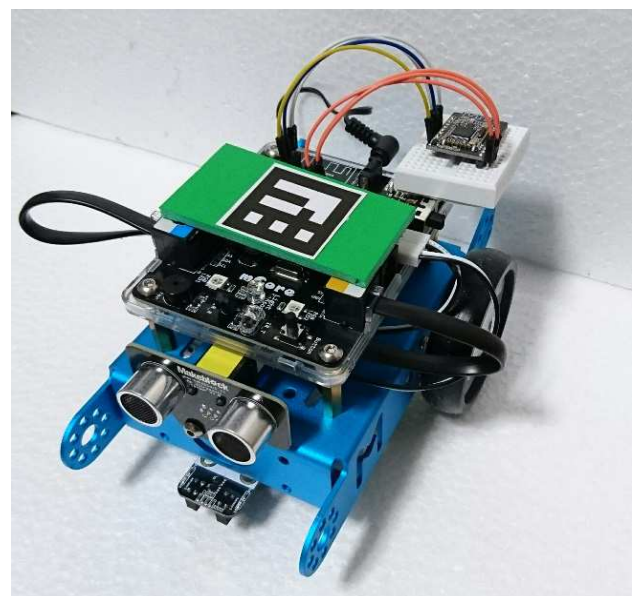


図 6 AR マーカーを付与したロボット教材

表1 本システムにおけるカードの種類

分類	動き	条件	音	変数	LED
機能	前進	もし壁に近づいたら	ドの音を出す	変数Aを定義	LED (赤) を点灯
	後退	ラインレース	レの音を出す	変数Bを定義	LED (青) を点灯
	停止	もしボタンが押されたら	ミの音を出す	A=0	LED (緑) を点灯
	右回り	もし明るかったら	ファの音を出す	B=0	LED (白) を点灯
	左回り	もし暗かったら	ソの音を出す	A+1	LED (黄) を点灯
	スピード (速い)	ループ	ラの音を出す	B+1	LEDを消灯
	スピード (普通)	違うなら	シの音を出す	A-1	
	スピード (遅い)	ここまでする	ド(高)の音を出す	B-1	
		0.5秒待機			
		1秒待機			
	2秒待機				

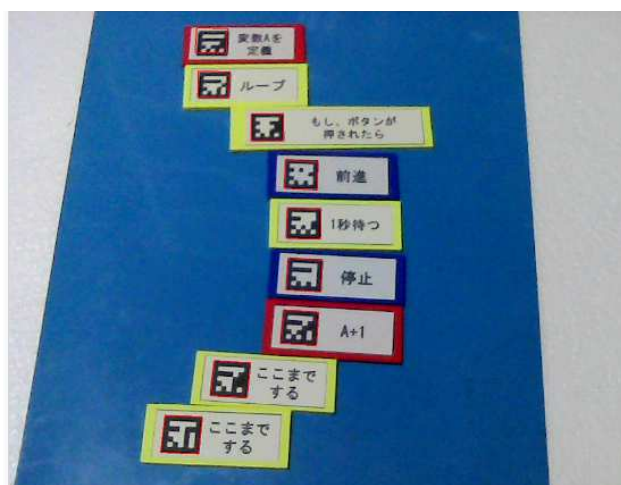


図7 カードによるプログラムの認識

読み取りプログラムの開発に Processing [16]と NyARToolkit (NyID マーカー) を用いたが、これは同一のマーカーを複数同時に認識できないため、同じ処理を表すカード同士でもマーカーの異なるものを複数用意した。

本システムは、まず PC のカメラで各カードのマーカーを検出する。プログラムによってカードが認識されると PC 画面上では図7に示すようにマーカー部分を囲む赤い枠が表示されるので、すべてのマーカーが認識されるように撮影すると、システムにプログラムが読み込まれる。なお、図7の例では、分かりやすいようにインデント (段差) をつけてカードを並べているが、プログラムとしての認識ではカードの上下関係のみを用いており、Arduino のソースコードと同様にインデントを崩してしまっても同じように動作する (制御ブロックは、閉じかっこに対応する「ここまでする」カードで閉じる)。

本試作システムの実装では、カード1枚がほぼ Arduino のコード1行に対応するので、カメラ映像から認識されたマーカーの配置から、単純に各マーカーを上から順に1行ずつ Arduino のコードに変換している。システムはそれをプログラムのテキストファイルとして、いったん PC に保

```
#include <MeMCore.h>
#include <SoftwareSerial.h>
MeDCMotor motor1(M1);
MeDCMotor motor2(M2);
MeLineFollower lineFinder(PORT_2);
MeUltrasonicSensor ultraSensor(PORT_3);
MeRGBLed rgb(0, 16);
MeLightSensor lightSensor(PORT_6);
MeBuzzer buzzer;
SoftwareSerial bluetooth (0, 1);
int speed = 120;
int A;

void setup() {
  bluetooth.begin(115200);
  pinMode(A7, INPUT);
  rgb.reset(PORT_7, SLOT2);
  rgb.setColor(0, 0, 0); rgb.show();

  bluetooth.println(¥"Start¥");
}

void loop() {
  if((0 ^ (analogRead(A7) > 10 ? 0 : 1))) {
    motor1.run(-speed); motor2.run(speed);
    delay(1000);
    motor1.stop(); motor2.stop();
    A += 1;
    bluetooth.print(¥"+A¥"); delay(100);
  }
}
```

図8 変換後の Arduino プログラムの例

存する。その過程で、プログラムの変数の値を PC に送信するコードがシステムによって付加される。図 8 は図 7 のカード配置によるプログラムが変換された Arduino のプログラムの例である。基本的に loop 関数の中身がほぼ 1 カード 1 行に対応しており、変数の値が変化する部分には、Bluetooth への出力が挿入されていることが分かる。

最後に、学習者は変換されて保存された Arduino のプログラムファイルを Arduino IDE に読み込んでコンパイルし、さらに PC とロボットのマイコン端子を USB ケーブルで接続して、コンパイルしたプログラムを転送する（この手順は、現在のところはまだ手動である）。ロボットに読み込まれたプログラムは、ロボットを起動することによって実行開始する。

4.3 プログラム実行状態の AR 表示の実現

ロボットの動作中に、学習者がそれに付与されている AR マーカーを PC のカメラで撮影すると、その情報が PC のプログラムで処理され、図 9 の例のように、PC の画面上でプログラム実行中のデータ（変数の値）がロボットの映像に重ねて AR 表示される。残念ながら、現在のところは表示されるデータ（プログラム中で使用できる変数）は変数 A と変数 B に固定されている。

この機能を実現するため、AR マーカーの認識と AR による情報表示には、タンジブルプログラミング機能における処理命令のカードの認識と同様に、Processing と NyARToolkit を使用している。

本システムの特徴は、AR 表示されるデータが Bluetooth 通信によってリアルタイムにロボットから PC に送信され、それにもなってリアルタイムに表示が更新されることである。AR を利用した既存の多くの教育システムは、コンピュータで生成した情報の提示方法としてのみ、AR を用いるものが多いが、我々は現実空間の教材から得たリアルタイムな実測情報を AR で提示するという双方向的な手法に着目して研究を進めている。

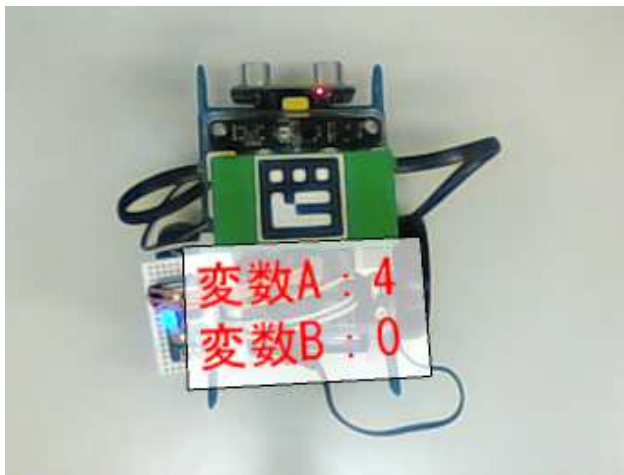


図 9 AR によるプログラム変数の値の表示

5. 考察と今後の展望

現在のシステムは、変数が整数型のもの 2 個しか使えず、それらも 0 による初期化と 1 ずつの増減しかできないなど、機能面で十分とはいえず、まだ完成とはいえない。しかしながら、現在の機能だけでも、ロボットを楕円のコースに沿って走らせるラインレースのプログラムは実現することができた。このことから、初歩的なプログラミング学習のための最低限の機能は備えていると考えている。

本システムを数人の大学生に見せて実際に使ってもらって見たところ、特にプログラミングではカードの並べ方を巡って活発な議論が生まれ、狙い通りタンジブルプログラムの採用によって協同学習が促されることが示唆された。また、用意された機能が少ないため、かえって通常のプログラミングよりもパズル的な面白みも感じられた。

AR の利用は、変数の値がリアルタイムに変化するので動作確認に有効であったが、変数名が A と B に固定されているためにプログラムの中での役割が分かりづらく、AR としての利点を十分に活かせていない結果となった。

以上のような反省に基づいて、現在、システムの改良を進めている。その後、実際にプログラミングを学び始めたばかりのユーザに使ってもらうことで、教材としての有用性について評価したいと考えている。

システムの完成に向けての改善点としては、(1) 変数の個数や種類を増やし、任意の名前をつけられるようにする、(2) 四則演算など処理命令も増やす、(3) PC 側での AR 表示を改善し、よりグラフィカルに可視化する、(4) プログラムの読み取りからロボットへの転送までの処理を自動化する、(5) 簡単な文法エラーには、メッセージを出すようにする、というようなことを優先的に取り組んでいきたい。

我々は、本研究を、教育用ロボットだけでなく、ハードウェアと連携するソフトウェアに対するデバッガの機能を AR によって実現する技術につながると考えている。

6. おわりに

本論文では、小型のロボットを用いたプログラミング教材において、プログラム実行中の変数の値などのシステムの内部状態を AR 技術によって可視化することで、学習者の理解を助ける手法について提案し、現在開発中のシステムについて述べた。プログラミング言語としては、初級者でも物理的なカードを配置することによってプログラミングを行えるタンジブルプログラミングを採用した。

本試作システムは、まだ機能的に不十分な部分が多く、完成しているとはいえないが、今後改良を進め、完成後には実際にプログラミング初級者（できれば子どもたち）に使ってもらうことで教材の有用性について評価したい。

謝辞

本研究の一部は、科研費 JP18K02907 の助成を受けたものである。

参考文献

- [1] 石塚丈晴, 堀口龍也: 英国の公立小学校における教科「Computing」におけるプログラミング教育の内容, 情報処理学会 コンピュータと教育研究会, Vol. 2015-CE-131, No. 12, pp. 1-4, 2015.
- [2] 文部科学省, 小学校プログラミング教育に関する概要資料, 2016. http://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afieldfile/2018/03/30/1375607_01.pdf
- [3] Lifelong Kindergarten Group (MIT Media Lab): Scratch, 2006. <https://scratch.mit.edu>
- [4] D. Bau, J. Gray, C. Kelleher, J. Sheldon, F. Turbak: Learnable Programming: Blocks and Beyond, Communications of the ACM, Vol.60, No.6, pp.72-80, 2017.
- [5] 鈴木栄幸, 加藤浩: 共同学習のための教育ツール「アルゴブロック」, 認知科学, No.2, Vol.1, pp.36-47, 1995.
- [6] Makeblock: mBot, 2016. <https://www.makeblock.com/steam-kits/mbot>
- [7] 玄人志向: KR-MBBOT, 2018. http://www.kuroutoshikou.com/product/others/others_iot/kr-mbbot/
- [8] Kitronik: :MOVE mini buggy kit, 2017. <http://www.kitronik.co.uk/5624-move-mini-buggy-kit-excl-microbit.html>
- [9] 八城明仁, 原田泰, 迎山和司: タンジブルなプログラミング学習環境の要件の検討, 情報処理学会 情報教育シンポジウム SSS2015, pp.207-213, 2015.
- [10] Primo Toys, Cubetto: A toy robot teaching kids code & computer programming, 2018. <https://www.primotoys.com>
- [11] 塩澤秀和, 小松京平: マーカー型 AR とセンサーを用いた物理(回路・力学)実験教材の開発, 日本バーチャルリアリティ学会 第 61 回サイバースペースと仮想都市研究会, pp.27-32, 2017.
- [12] 廣田翔平, 塩澤秀和: AR 技術と加速度センサを利用したカード型の力学実験教材, 日本バーチャルリアリティ学会 第 63 回サイバースペースと仮想都市研究会, pp.35-38, 2018.
- [13] 塩澤秀和: AR 技術とセンサーを用いた物理実験教材の開発, 第 43 回教育システム情報学会全国大会, pp.435-436, 2018.
- [14] Arduino: Arduino - Home, <https://www.arduino.cc>
- [15] NyARToolkit project: NyARToolkit for Processing, <https://nyatla.jp/nyartoolkit/wp/?cat=5>
- [16] Processing Foundation: Processing.org, <https://processing.org>