

紙に対するインタラクションを活用した タンジブルなプログラミング環境

多田 一貴*¹ 田中 二郎*²

Tangible programming environment that take advantage of the interaction for paper

Kazuki Tada*¹ and Jiro Tanaka*²

Abstract – This paper introduces a tangible programming environment that uses paper cards as command objects. This system includes paper cards, a webcam, and software. The paper cards are printed descriptions and markers. The markers are captured by the webcam and are recognized by the software. The users can line up the command cards in a specific order to create a program. This system is capable of drawing and moving shapes, control some actuators (e.g. led and motor), and creates loops and branch executions. In addition, it is possible to edit the program by writing on the paper cards. For example, the number of loop executions and the values of any variables can be altered. The program can also implement draw commands for custom sketches designed by the users. We have performed simple evaluation experiments using this system and collected the opinions of the users.

Keywords : Interface Design, Programming Environment, Tangible User Interface

1. はじめに

義務教育のカリキュラムにプログラミングが取り込まれたことや、各地で小中学生を対象としたプログラミングワークショップが活発的に開催されていることにより、専門職以外の人々がプログラミングに触れる機会が日々増えつつある。そのため、初心者でも扱いやすいプログラミング言語の整備が求められるようになり、これまでに様々な環境の提案がなされてきた。Scratch^[1] や VISCUIT^[2] はその一例であり、これらは簡単な GUI 操作によってプログラムを作ることができる環境である。一方、プログラミング言語における命令文を実体のある物質的なものに置き換えた、いわゆる TUI(Tangible User Interface) を用いることによる環境の提案も多い。TUI をプログラミング環境に取り入れることは、GUI を用いた環境よりも学習に優位に働くことや、複数人で議論しながら学習を進める協調学習の効果があることが報告されている^[3]。

また、複雑な実行結果を短く容易な命令文によって実現可能とすることは、利用者のプログラミングへのモチベーション向上に寄与するという報告がある^[4]。

しかし、TUI を用いたプログラミング環境では、命令オブジェクトが有形であるが故にプログラム本体を保存することが難しく、人が物事を学習するにあつ

て重要なメモや試行錯誤の過程を残すことが難しい。

これらの現状を踏まえ、本研究では紙で出来た命令オブジェクトを使ったタンジブルなプログラミング環境を提案する(図1)。本環境は紙という実体のある命令オブジェクトを並べることによるプログラミング手法を実現し、命令オブジェクトへの書き込みや接触といった紙を用いたならではのインタラクションをプログラミングの過程に取り入れた。また、紙を用いることによりタンジブルなプログラムの保存が容易であり、メモやコメント文との共存が可能となる。

使用者はこの環境を用いたプログラミングによって、紙に試行錯誤の結果を残しながら、図形の描画や移動、音の再生、LED やモータなどアクチュエータの操作等を少ない命令を組み合わせることによって実現することができる。本論文では、この環境の概要、実装、特徴及び実施した被験者実験の結果と考察について報告する。

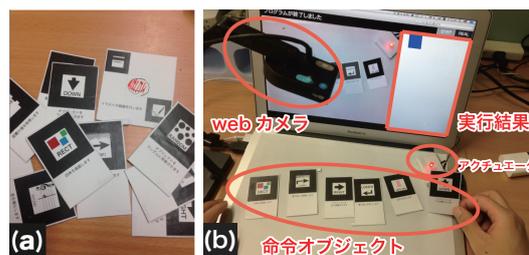


図1 (a) 紙媒体の命令オブジェクト (b) 環境の全体図

Fig.1 (a) Paper cards as command objects (b) System overview

*1: 筑波大学大学院 システム情報工学研究科

*2: 筑波大学 システム情報系

*1: Graduate School of Systems and Information Engineering, University of Tsukuba.

*2: Faculty of Engineering, Information and Systems, University of Tsukuba.

2. 関連研究

初学者に向けたプログラミング環境はこれまでに多く提案されてきた。GUI による操作が主な環境では、画面上の命令ブロックの組み合わせや、絵を描くことによってプログラミングを行う Scratch^[1] や、VIS-CUIT^[2] などがある。TUI を取り入れた環境では、木製の命令オブジェクトを組み合わせるロボットを制御する Tern^[3]、LED ブロックを用いた書き換え可能な命令オブジェクトを組み合わせる Tiles^[7]、Scrach の命令をブロックに置き換えた物質プログラミング^[6] などがある。紙で出来た命令オブジェクトを用いたものでは、オブジェクトを並べてキャラクタの動作をプログラムする Robert^[5] が存在する。

また、初学者に対してプログラミングに対するモチベーションを向上させるための研究として、簡単な構文で複雑な実行結果を実現する宮下の HMMML^[4] があげられる。

3. 紙に対するインタラクションを活用したプログラミング環境

3.1 概要

本論文が提案する環境は、紙媒体の命令オブジェクトを並べて組み合わせることによってプログラミングを行うことができる。紙媒体を命令オブジェクトとして採用することで、プログラミングの過程に紙に対する手書き入力や自由なスケッチを行うといったアナログな動作を組み込むことが可能である。ユーザは命令オブジェクトを組み合わせることでプログラムを作り、画面上における図形の描画・移動等や、実世界における LED・モータ等の操作を行うことができる。命令は、図や音、アクチュエータの操作、繰り返し、条件分岐、変数操作などが用意されている。

3.2 実装

本環境では、紙媒体の命令オブジェクト、web カメラ、Arduino をもとにしたアクチュエータセット、そして PC を使用する。紙媒体の命令オブジェクトには命令の内容を表すと共に認識のために使われるマーカと、命令の説明文が記述されている。並べられた命令オブジェクトは web カメラで撮影され、実行結果が PC の画面上またはアクチュエータの変化によって表れる。命令を認識し解釈を行った上で実行結果を反映するソフトウェアは ActionScript 及び openFrameworks によって実装した。

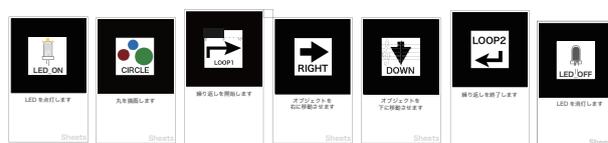
3.3 シンタックス

本環境は基本的に図形の描画や音の再生、アクチュエータの実行とその制御を組み合わせることがプログラミングを行う。そのため、各種図形を描画する命令である「描画命令」、任意の図形を移動させる「移動

命令」、サウンドを発生させる「音命令」、アクチュエータを操作する「アクチュエータ命令」、変数を操作する「変数命令」、繰り返しや条件分岐といった基本的なシンタックスを記述するための「構文命令」等が用意されている(図 2)。これらを組み合わせたプログラムの一例を図 3 に示す。



図 2 命令オブジェクトの例
Fig. 2 Example of command objects



LED を点灯し、丸を右下に 5 回移動させた後、LED を消灯する

図 3 プログラムの例
Fig. 3 Example of program

3.4 紙による命令オブジェクト

命令オブジェクトとして紙を用いることにより、紙がもつアナログな動作をプログラミングの過程に取り入れることができる。本環境では、命令オブジェクトへの書き込みによって繰り返し文の回数や変数の値を変更することができる(図 4(a))。LED の点灯色やモータの回転速度といったアクチュエータ操作のパラメータも、色ペンの使用や塗りつぶし動作によって操作できる。ユーザが書いた自由な絵を描画命令として扱うといったことも可能である(図 4(b))。

また、多くの命令で構成される複雑なプログラムをスキャンし、一つの命令オブジェクトにまとめる物理的な関数化が可能である。

その他大きな特徴として、本環境では作成したプログラムをノートや本に貼付けることができるため、周りに注釈や解説を書きながらの学習が可能である(図 5)。これは一般的なプログラミングにおけるコメント文に相当するものであり、ユーザの試行錯誤の過程を

アーカイブすることができるため、学習の大きな支えとなりえると考える。

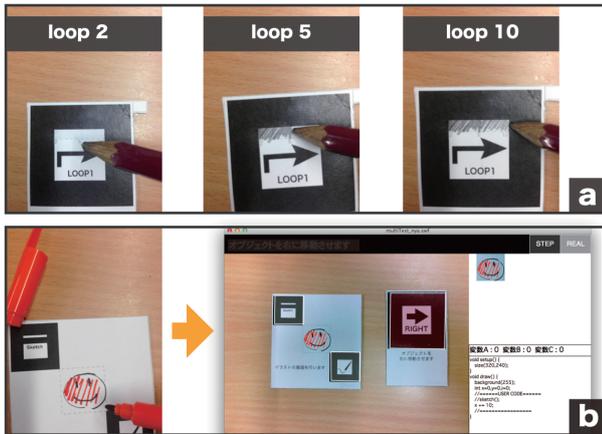


図 4 書き込みによるインタラクションの例
Fig. 4 Example of drawing interaction

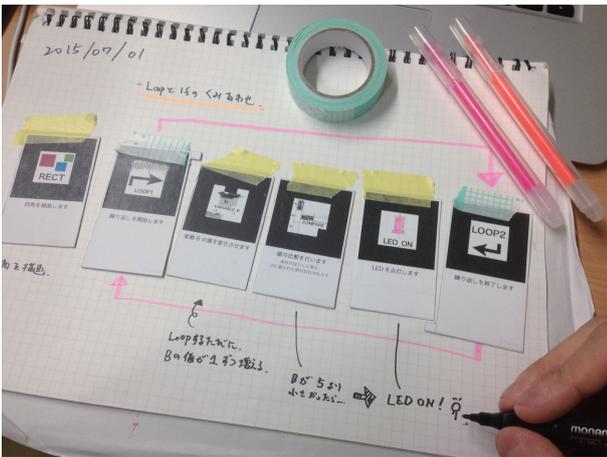


図 5 プログラムをメモとともに保存する例
Fig. 5 Example of archives program with annotations

4. 評価

4.1 実験内容

本環境のユーザビリティ及び学習効果を評価するために、被験者実験を行った。被験者は18歳から23歳までの合計12名(女性3名)であり、全員が大学生もしくは大学院生である。プログラミングを日頃から使用する情報系の学生が7名、授業等で多少の経験がある学生が3名、全く経験のない学生が2名であった。被験者は本環境を用いたプログラミング方法や特徴の説明を受けた後、いくつかの問題が提示され、それに回答する形でプログラミングを行う。また、問題回答形式の実験の後、自由にプログラミングを行う時間を設けた。実験が行われている間、我々は被験者が問題に対して正しいプログラムを作るまでの時間と間違ったプログラムを作った回数の計測、及び被験者の行動

観察を行った。また、実験終了後、被験者に対し、環境についての操作性、思った通りのプログラミングが行えたか、紙媒体を用いていることへの感想、総合的な好感度等について7段階のリッカート尺度を用いたアンケートを行い、その集計を行った。

実験内容は紙へのインタラクションが含まれた環境のものと、含まれていない環境の2通りを用意し、被験者はどちらか片方のみを体験した。提示される問題は同じ物であるが、紙へのインタラクションが含まれていない環境においては、図4(a)で示したようなループ回数や変数値、条件分岐などの指定を行う際に、あらかじめ塗りつぶして用意されたものの中から適切なものを選ぶといった、紙に対する塗りつぶしや書き込みなどの動作が制限された条件で実験が行われた。

この2つの異なる条件下における被験者の環境への評価を比較することにより、紙へのインタラクションがプログラミングの過程に取り組みれることが、プログラミングへのモチベーションや環境への好感度に影響するの測定を行った。

4.2 結果

表1は被験者の問題に対する回答時間と間違ったプログラムを作成した回数をまとめたものである。回答時間においては、被験者のこれまでのプログラミング経験度によって大きく差が観測されたが、プログラミング経験度による誤回数はほとんど差が生じなかった。このため、本環境における命令シンタックスが未経験者においても容易に理解できるものであったと考えられる。

表1 被験者のプログラミング経験度による間に対する平均回答時間と平均誤回答数(sec)

Table 1 The average error number and average response of answer by the programming experience of the participants.

経験度	平均回答時間(秒)	平均誤回答数(回)
日常的に使用	66.07 (SD=18.89)	0.5 (SD=0.61)
授業程度	133.33 (SD=57.74)	0.67 (SD=0.47)
未経験	210 (SD=50)	0.5 (SD=0.5)

また、紙へのインタラクションが含まれた環境(GroupA 7人)と、含まれていない環境(GroupB 5人)における、被験者の実験後アンケートの平均評価値を示したものが表2となる。評価値は7段階のリッカート尺度によるアンケート調査によって集計された物である。項目は、環境の操作性(使いやすいかどうか)、紙媒体使用の印象(親しみやすいかどうか)、思い通りの操作(自分の意図した通りのプログラミングができるかどうか)、環境への好感度(この環境が好きになるかどうか)、使用モチベーション(使用していて楽しさやわくわく感が生まれるかどうか)の5つである。

表2 アンケート調査の結果

Table 2 The results of the questionnaire survey.

	GroupA	GroupB
環境の操作性	6 (SD=1.06)	6 (SD=0.63)
紙媒体使用の印象	6.71 (SD=0.45)	5 (SD=1.09)
思い通りの操作	6.28 (SD=0.70)	5.4 (SD=0.8)
環境への好感度	6.86 (SD=0.35)	5.6 (SD=0.8)
使用モチベーション	6.57 (SD=0.50)	5.4 (SD=1.02)

すべての項目において、紙へのインタラクションが含まれた環境 (GroupA) への評価値が、紙へのインタラクションが含まれていない環境 (GroupB) への評価値と同等かそれ以上のものとなった。また、被験者毎のデータを用いた t 検定においては、紙媒体使用の印象と環境への好感度について、紙へのインタラクションが含まれた環境への評価値の方が有意に大きいという結果となった (紙媒体使用の印象 $p = 0.023 < 0.05$, 環境への好感度 $p = 0.017 < 0.05$)。以上の結果から、タンジブルなプログラミング環境においては、タンジブルなオブジェクトを整列して命令を組み上げることに加えて、ユーザにとって身近な動作をプログラミングの過程に取り込むといったインタラクティブ性を付与することで、ユーザのプログラミング環境に対する印象が明るくなり、親しみ易く感じるようになることが分かった。

その他、アンケートの他に実施した口頭インタビューからは、「鉛筆などアナログなものを使ったため、プログラミングを身近に感じることができた」、「塗り潰しなどの動きがあり面白い」といった感想を得られた。しかし一方で、「初学者はこれで十分かもしれないが慣れた人には物足りなく感じる」、「実行結果はもっと大きく、激しいほうが楽しい」といった意見も得られたため、用意する命令群や出力される実行結果、作成可能なプログラムの規模等に改善が必要であると考えられる。

5. まとめ

本論文では紙媒体の命令オブジェクトを用いたタンジブルなプログラミング環境について述べた。本環境では、ユーザは紙に対する書き込みや接触といったインタラクションを行いつつ、短い命令で図形の操作や LED やモータなどのアクチュエータを操作するプログラミングを行うことができる。命令オブジェクトが紙製であるため、ノート等に貼付けてプログラムの保存が可能であり、命令の周りに注釈や解説を残すことにより、学習の過程を後々振り返ることができる。被験者実験においては、本環境の扱いやすさや親しみやすさを計測し、初学者においても扱いが容易な環境であ

ることを示した。また紙へのインタラクションをプログラミングの過程に取り込むことが学習に有益であることを示した。

今後は命令オブジェクトのデザインや内容、作成可能なプログラム規模の拡大についての考察、改善を行うと共に、既存環境との比較評価や長期的な学習効果の測定を行う。

参考文献

- [1] Mitchel, R., John, M., et al.: *Scratch programming for all*; Communications of the ACM, **Vol.52**, No.11, pp.60-67, (2009).
- [2] Harada, Y. and Potter, R.: *Fuzzy rewriting: soft program semantics for children*; In IEEE Symposium on Human Centric Computing Languages and Environments, **Vol.1**, No.1, pp. 39-46, (2003).
- [3] Horn, S.M., Solovey, T.E., et al.: *Comparing the use of tangible and graphical programming languages for informal science education*; In Proceedings of the 27th international conference on Human factors in computing system (CHI'09), pp.975-984, (2009).
- [4] 宮下: プログラミングに対するモチベーションを向上させる新言語 HMMMML の開発; 第 51 回プログラミングシンポジウム論文集, pp.57-64, (2010).
- [5] Horn, S.M., AlSulaiman, S., et al.: *Translating Roberto to Omar: Computational Literacy, Stickerbooks, and Cultural Forms*; In Proceedings of the 12th International Conference on Interaction Design and Children (IDC'13), pp.120-127, (2013).
- [6] 八城, 迎山: 物質プログラミング -物質によるプログラムの可視化と開発環境の制作-; インタラクション 2014, pp.647-650, (2014).
- [7] 川崎, Nguyen, 秋田: 2色での入力・表示可能なマトリクス LED ブロックを用いたタンジブルなプログラミング教材; インタラクション 2014, pp.206-209, (2014).