

# Scratchのプログラミング学習における 画像処理技術を用いた採点支援システムの開発

## Development of Grading Support System Using Image Processing for Scratch

土屋悠斗<sup>†</sup>      柴田怜<sup>†</sup>      田面大輝<sup>†</sup>      千々岩水葵<sup>†</sup>      富田真光<sup>†</sup>      西川竜雅<sup>†</sup>  
 Yuto Tsuchiya    Ren Shibata    Daiki Tanabo    Mizuki Chijiwa    Masamitsu Tomita    Ryuga Nishikawa  
                          池辺正典<sup>†</sup>      佐野昌己<sup>†</sup>      櫻井淳<sup>†</sup>  
                          Masanori Ikebe      Masami Sano      Jun Sakurai

<sup>†</sup> 文教大学 情報学部

<sup>†</sup> Faculty of Information and Communications, Bunkyo University.

### 要旨

2020 年度に必修化された小学校のプログラミング教育では、プログラミング的思考を育むために、各教科などにプログラミング体験を取り入れることを狙いとしている。こうした背景から、児童を対象としたプログラミング学習を実施するパソコン教室が注目されており、ビジュアルプログラミング言語の Scratch が広く活用されている。ここでは、プログラミング検定に準拠した模擬試験を用いて講師が手動で採点しているが、作業負担や採点ミスが生じる問題がある。そこで、本研究では、画像処理技術を用いて Scratch の採点作業を支援するシステムを開発する。

## 1. はじめに

2017 年 3 月に文部科学省より公示された小学校の新学習指導要領[1]において、2020 年度より小学校のプログラミング教育が必修化された。こうした背景から、児童を対象としたプログラミング学習を実施するパソコン教室が注目されており、ジュニア・プログラミング検定などで採用されるビジュアルプログラミング言語の Scratch が広く活用されている。ここでは、検定に準拠した模擬試験を用いて講師が手動で採点しているが、作業負担や採点ミスが生じる問題がある。こうした問題に対応するために、Scratch を対象にブロック属性[2]やソースコード[3]を用いて自動で採点する研究が行われているが、文字情報を利用しているために複雑なコードに対応できない問題が潜在する。そこで、本研究では、Scratch の画面上で実行される動作に着目し、画像処理技術を用いて、作成したコードの正誤を判定するシステムを開発する。また、その精度を検証し、パソコン教室への導入の可能性を検討する。

## 2. システム概要

本研究では、パソコン教室に向けたプログラミング学習の採点支援システムを開発する。開発システムの概要を図 1 に示す。処理の流れとして、まず、解答データ生成機能では、システム上で模範解答と採点対象の Scratch 形式ファイルをそれぞれ読み込む。そして、pyautoGUI を用いたキーボードの自動制御により Scratch のコードを自動で実行し、その実行画面を 1 秒間に 3 フレームの間隔で画像として生成する。また、一連の動作における音声ファイルも生成する。次に、動作の判定機能では、画像処理ライブラリである OpenCV を用いて、模範解答と採点対象の対応するフレームの画像を順番に比較し、同じ動作であるかを判定する。そして、すべて一致した場合は正解、不一致の場合は不正解として最初に不一致であったフレームを記録する。さらに、音の判定機能では、numpy のライブラリを用いて、音声ファイルを波形グラフの画像に変換し、模範解答と採点対象の画像を比較して同じ音であるかを判定する。これらの処理により、正解または不正解の判定結果を出力する。

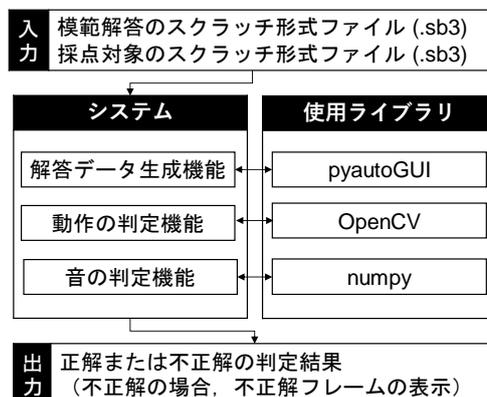


図 1 開発システムの概要

### 3. 評価実験

#### 3.1. 実験概要

評価実験では、Scratch の模擬試験により作成したファイルを用いて、提案手法による正解と不正解の判定精度を検証する。実験手順として、まず、模擬試験として、ジュニア・プログラミング検定に準拠したエントリー級・ブロンズ級・シルバー級・ゴールド級の4つの問題を作成した。例として、エントリー級の問題の抜粋を図2に示す。次に、それらの模擬試験に対して、それぞれ模範解答ファイルを作成した。そして、解答の正解または不正解の判定用として、ソースコードの異なる採点対象ファイルを正解と不正解でそれぞれ6ファイルずつ作成した。例として、正解の解答コードの2例を図3に示す。ここで示した2例は、ブロックの個数や配置などのコード上は異なるが、実行画面の動作は同様であり、模擬試験としては両方正解となる。このように、Scratch のコードが不一致の場合においても、正解または不正解を正しく判定できるかを評価した。

#### 3.2. 実験結果

実験結果を表1に示す。この結果から、不正解ファイルは正しく判別できているものの、正解ファイルの一部が正しく判別できていないことがわかる。誤判定の原因として、解答データ生成機能において、画面録画の開始後、自動制御のキー入力で行う実行開始のボタンをクリックする際に、時間のラグが生じる影響により、比較対象のフレームの番号が一致しない場合があることがわかった。この問題に対しては、判定処理の開始タイミングに関して、録画開始時ではなく、ボタンのクリック時に設定することで対処することを検討している。

### 4. おわりに

本研究では、パソコン教室に向けて、画像処理技術を用いてScratchの採点作業を支援するシステムを開発し、ジュニア・プログラミング検定の模擬試験の判別精度を検証した。今回は正誤判定のみに限定したが、今後は各配点で点数化できるアルゴリズムに改良したい。

### 5. 参考文献

[1] 文部科学省：小学校学習指導要領，2017。  
 [2] 畠中明哉，尾崎剛，広瀬啓雄：Scratchドリルシステム開発に向けたScratchプログラム移動採点方法の検討，教育システム情報学会，2022年度学生研究発表会，pp.29-30，2022。  
 [3] 若松玲依，榎原絵里奈，小野景子，新濱遼大：Scratchを用いたプログラミング演習における教員支援を目的とした採点支援システムの提案，ソフトウェア工学の基礎ワークショップ，pp.49-54，2022。



図2 模擬試験の問題抜粋（エントリー級）

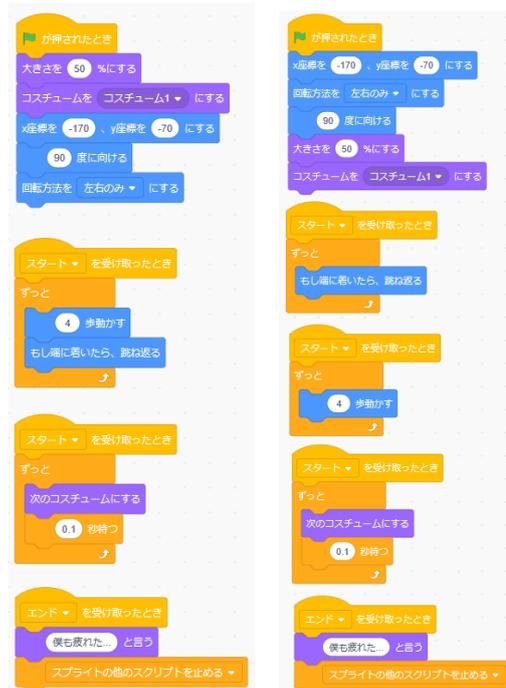


図3 正解の解答コード2例（エントリー級）

表1 実験結果

対象	正解ファイル			不正解ファイル		
	作成数	正解判定数	一致率	作成数	不正解判定数	一致率
エントリ	6	3	50%	6	6	100%
ブロンズ	6	2	33%	6	6	100%
シルバー	6	5	83%	6	6	100%
ゴールド	6	2	33%	6	6	100%