

オブジェクトの動作に基づく Scratch 作品の直観的検索手法

ソーシャルソフトウェア工学研究室

60236238

福地 ユキ

概要：ビジュアルプログラミング言語 Scratch において、ユーザはオンラインサービス上に公開された他者の作品をキーワードで検索し、参照することで実装方法を学習する。しかし、ユーザの持つ動作のイメージをキーワードに変換することは難しく、作品のタイトルや説明文に動作のキーワードを含まない作品は検索結果に表示されないため、作品検索は容易ではない。本研究では Scratch 作品の検索を容易にするために、キーワードによる検索ではなく動作のイメージを入力とする直感的な検索手法を提案する。

キーワード：Scratch, コード検索, 時系列解析, 物体追跡, 画像検索

1 はじめに

ビジュアルプログラミング学習サービスの Scratch では、ユーザが制作したプログラム作品をオンラインサービス上に公開可能である。オンラインサービス上には、膨大な数の作品が公開されており、ユーザは他者の作品を参照することで多様な実装方法を学習する [1]。参照する他者の作品を探し出すためにキーワード検索を行うが、ユーザがイメージする動作を言語化することは難しく、動作に関するキーワードをタイトルや説明文に含まない作品は検索不可能であるため、ユーザがイメージする動作を含む作品を検索することは容易ではない [2]。動作のイメージを入力とする検索を実現することで、作品検索を容易にできると考える。本研究では、マウスを用いた描画による直観的な操作で動作のイメージを入力し、入力と検索対象作品に含まれるオブジェクトの座標変化データの距離を算出することで Scratch 作品を検索する手法を提案する。

2 オブジェクトの動作に基づく Scratch 作品検索手法

2.1 手順 1: オブジェクトの座標変化を取得

手順 1 では、入力の動作イメージと検索対象作品に含まれる動作の座標変化データを取得する。

2.1.1 動作イメージの入力

ユーザの持つ動作のイメージをマウスで描画することによって入力する。動作のイメージを描画中、0.2 秒ごとにマウスの座標を取得する。取得した座標を入力座標変化データとする。

2.1.2 検索対象作品に含まれるオブジェクトの動作

検索対象作品の収集 Web アプリケーションの自動テストツール Selenium¹ を用いて、検索対象作品の実行結果のスナップショットを 0.2 秒ごとに収集する。

画像認識による座標変化データ取得 収集した各作品のスナップショットに含まれるオブジェクトの座標位

置を、SIFT(Scale-Invariant Feature Transform) を用いた画像認識によって取得する。Scratch では、オブジェクトの色変更や回転・拡大縮小を自由にできることから、そのような変更を行なった作品にも対応するため、照明変化や回転・拡大縮小に頑健である SIFT を採用する。

2.2 手順 2: 座標変化同士の距離を算出

座標変化データの正規化 入力と検索対象動作の座標が離れていても、動作の軌跡が類似していれば類似動作として抽出を行うため、手順 1 で取得した時系列データを最小値 0、最大値 1 に正規化する。

DTW 距離の算出 入力と各動作の座標変化の類似性を測るために、2 つの時系列データの距離を算出する動的時間伸縮法 (DTW: Dynamic time warping) によって距離を算出する。異なる長さの時系列データ同士でも距離算出が可能であるため、DTW を採用する。

3 類似動作を抽出可能な距離の分析

類似動作を抽出可能な距離を明らかにするために、複数の入力を行い、DTW 距離の算出結果について定量的分析と定性的分析を行う。また、提案手法による作品検索の精度を明らかにするために、ランキング指標を用いた検索評価を行う。

3.1 検索データセット

Aivaloglou らの公開データセット²のうち、13,437 件の Scratch 作品を検索対象とする。

3.2 入力

直線移動や曲線移動をする 4 つの動作を入力する。

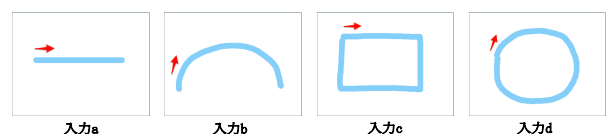


図 1: 分析で用いる 4 つの入力

¹<https://www.selenium.dev/>

²<https://github.com/TUDELFTScratchLab/ScratchDataset>

3.3 定量分析

移動方向が入力と異なる動作が抽出される距離を明らかにするために、各入力に対する距離算出結果を昇順に並べ、ある2点の傾きが収束する値を求める。傾きが0.5未満となり、動作を目視で確認した結果移動方向が入力と異なっている場合、その時点の距離を収束する値とする。各入力に対して定量分析を行なった結果を表1に示す。

表 1: 移動方向が異なる動作が抽出される距離

入力	距離
a	4.4
b	3.2
c	6.9
d	7.9

3.4 定性分析

類似動作を抽出可能な距離を明らかにするために、入力と各検索対象動作の DTW 距離算出結果から距離の異なる複数の動作を提示し、動作の類似性をアンケートで評価する。動作の類似性は、4段階のリッカート尺度 (1: 類似する, 2: やや類似する, 3: やや類似しない, 4: 類似しない) で評価し、初めて回答者全員が「4: 類似しない」を選択した距離未満を、類似動作を抽出可能な距離とする。「1: 類似する」以外を選択した場合、理由を記述する。アンケートには、3.3で明らかになった移動方向が入力と異なる動作が抽出される距離までの動作を提示する。各入力における類似動作を抽出可能な距離の結果を、表2に示す。

表 2: 類似動作を抽出可能な距離

入力	距離 (未満)
a	4.3
b	2.2
c	4.1
d	4.5

3.5 検索評価

本研究で提案する検索手法の精度を評価するために、各入力に対する距離算出結果のうち、距離の近い上位20件を用いて3.4と同様にアンケートを実施した。検索評価のために、MRR(Mean Reciprocal Rank)を算出するMRRは、ランキング中に初めて正解が出現した順位の逆数を全ての入力について平均したランキング指標である。アンケートの結果から、初めて回答者全員が質問1に対して「1: 類似する」を選択した動作を正解とし、MRRを算出した結果0.88となった。

4 考察

4.1 動作に基づく検索手法の実用性

3.4で実施した定性分析の結果から、見た目と実際の座標変化に差異がある動作が存在し、その原因が画像認識にずれがあることや、移動方向は類似するが曲線(直線)移動ではなく直線(曲線)移動をする動作が抽出され、その原因が座標変化取得の時間間隔にあることから、検索の精度向上のために改善が必要であることが明らかとなった。しかし、いずれの入力においても回答者全員が類似すると判断する動作が複数存在しており、??の検索評価の結果からランキング上位に類似動作が出現することが期待できるため、動作に基づく作品検索手法は実用性があると考えられる。

4.2 抽出した類似動作のプログラム

動作に基づく作品検索を行うことで、類似動作の多様な実装方法を検索可能であることが明らかになった。実装方法の異なる類似動作を提示することは、プログラムの実装方法やより品質の良い実装の両方の学習に繋がり、特に初学者も多く利用しているScratchにおいて有用であると考えられる。

5 おわりに

本研究で提案した手法により、動作のイメージから容易に作品を検索することができるため、他者の作品を参照することによる実装方法学習の支援を期待する。さらに手法を改善することによって、より精度の高い作品検索の実現を期待する。

参考文献

- [1] Mitchel Resnick, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Saul Silver, Brian S Silverman, Yasmin Bettina Kafai, Scratch: Programming for all, Communications of the ACM, Vol.52, No.11, pp.60-67, 2009.
- [2] Aniket Dahotre, Yan Zhang, Christopher Scaffidi, A qualitative study of animation programming in the wild, ESEM'10: Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, No.29, pp.1-10, 2010.