

ペイントソフトにおける 操作ログに基づいた機能の推薦

三島 優理菜[†] 武井彩佳[§] 山名 早人[‡]

[†]早稲田大学基幹理工学部 〒169-8555 東京都新宿区大久保 3-4-1

[§]早稲田大学大学院基幹理工学研究科 〒169-8555 東京都新宿区大久保 3-4-1

[‡]早稲田大学理工学術院 〒169-8555 東京都新宿区大久保 3-4-1

E-mail: {mishima3456, ayaka, yamana}@yama.info.waseda.ac.jp

あらまし 近年、デジタル機器の普及が進み、専門家だけでなく様々な人がデジタル機器を利用できるようになった。今までペンや絵の具を使って紙に絵を描くアナログな絵を描いていた人が、PC を使ってデジタルで絵を描くとなると、そこには大きな溝が生じてしまう。これは、アナログとデジタルでは、使う道具、描き心地、描く過程、必要な知識などかなりの相違があるためである。特に、デジタル機器において画像を扱うという特性上、ペイントソフトは多くの複雑な機能を有しており、短時間で上手く使いこなすことが難しい。また、ブラシと消しゴムと色の選択さえできればある程度の絵は描けてしまうため、便利な機能に気づかないまま作業を行ってしまうという問題がある。気づかないまま作業を行ってしまい、機能を使えば短縮できた時間を長い期間に渡って無駄にしてしまう可能性がある。そこで、これらの問題を解決するため、本研究では、ペイントソフト初心者を対象とし、ユーザの操作ログからペイントソフトが持つ機能を推薦する手法を提案する。提案手法では、絵を描く基本作業が、「線を引き」ことと「色を塗る」ことであることに着目し、SVM を用いて操作ログを「線を引き作業」と「色を塗る作業」に分類する。そして、各々の作業に対してペイントソフトが持つ機能を用いていなかった場合、利用者に対して当該機能を推薦する。ペイントソフトでは、「線」と「塗り」の工程において、利用できる機能が異なることから、この二種類の作業を分類できれば、適切な機能推薦ができると考えた。評価実験では、予め指定された絵を描いてもらい、推薦された機能についてのアンケートを取った。16 人での評価実験の結果、ユーザが知らなかった機能でかつ便利だと感じる機能の推薦が 60%の割合で行われた。

キーワード ソフトウェア機能推薦, ペイントソフト, 操作ログ

1. はじめに

近年、PC やスマートフォンなど、多くのデジタル機器が一般に普及し、こうした機器に対する敷居が低くなっている。そして、これらのデジタル機器を使って絵を描くことが行われるようになった。PC、ペンタブレット、そしてペイントソフトを揃えるだけで、手軽に絵を描くことができる。

デジタル機器の普及に伴い、デジタルで絵を描くことへの敷居も低くなっており、SNS や Pixiv¹のようなイラストコミュニケーションサービスに、デジタルで描いた自分の絵を投稿することが盛んになっている。しかし、今までペンや絵の具を使って紙に絵を描くアナログな絵を描いていた人が、PC を使ってデジタルで絵を描くとなると、そこには大きな溝が生じる。これは、アナログとデジタルでは、使う道具、描き心地、描く過程、必要な知識などかなりの相違があることに起因する。すなわち、たとえアナログで絵が上手な人であっても、すぐにデジタルで上手な絵が描けるわけではない。デジタル機器において画像を扱う際に重要となるのは、

「アナログとデジタルの違い」と「ペイントソフトは多くの複雑な機能を有しているが、すぐに上手く扱うのは難しい」という点である。アナログとデジタルでは絵の質感などを表現する方法に大きな違いがあるため戸惑いが生じることがある。また、ペイントソフトが提供する機能を使いこなせないと、モチベーションが下がってしまったり、創造性を発揮できなかつたりという問題が生じてしまうことがある。特に、便利な機能を知らずに作業することで、無駄な作業をしてしまうことにもつながる。ペイントソフトにおいては、最低限の機能である色の選択とペンツールと消しゴムツールを知っていればある程度のものは描けてしまうため、便利な機能を知らないままソフトウェアを使い続けてしまう。

このような、アナログとデジタルの違いと、ソフトウェアの機能を知らずに作業を進めてしまうという問題点を解決するため、ペイントソフトの機能を推薦することが有効であると考えられる。

グラフィックを扱うソフトウェアを対象とした機能推薦に関する従来研究には、(1)大量の他ユーザのデ

¹ <https://www.pixiv.net/>

ータを用いた推薦システムアルゴリズムを用いた手法と、(2) ウェブ文書を用いる手法がある。

大量の他ユーザのデータを用いた推薦システムアルゴリズムを用いた手法として、設計支援ソフトウェアとして一般的に使われる AutoCAD を用いた研究[1][2][3]があり、開発元の Autodesk 社 がユーザから大量にデータを集め、それを推薦システムのアルゴリズムに適用し、機能を推薦している。

ウェブ文書を用いた手法は、ペイントソフトとしても使われるフリーソフトの GIMP を用いた研究[4]である。この研究では、ウェブ文書から機能と作業をマッピングし、そのマップである QF-Graph を用いて機能を推薦している。マッピングの結果により、機能と作業の関連性を 0~1 の重みで表現する。ユーザが最近使った一定数の機能を指定し、その重みに基づいて作業を選択する。そして、その選択された作業から、重みに基づいて選択した作業との関連度が高い関連する機能を推薦するという手法である。

以上紹介した手法にも問題点が存在する。Autodesk 社の研究[1]では、作業と機能を関連付けた推薦ができないため、機能について知っても使う場面が分からない。ウェブ文書を用いた研究[4]では作業と機能をマッピングすることにより、その問題点を解決している。しかし、作業の推定に用いるのはコマンドだけであるため、正確に作業を推定することが難しい。

そこで、本研究では、ペイントソフト初心者を対象とし、ペイントソフトの操作ログを取り、当該ログを元に絵を描くための機能を推薦する手法を提案する。アプリケーションはオープンソースのフリーソフトである Krita²を用いる。本研究では、便利な機能に気づかないまま作業を行い、長い時間をかけてしまう問題を解決することを目的としているため、ペイントソフトを使い始めて時間が経っていない初心者を対象とした。

提案手法では、まず、得られたログデータを、SVM を用いて「線」と「塗り」の過程に分類する。そして、線と塗りの過程においてあらかじめ決めておいた推薦する機能について、それぞれの過程で使われているかどうかにより推薦する。線と塗りに着目した理由は次の通りである。絵を描くことにおいて、もっとも基本となる作業が「線を引く」と「色を塗る」ことである。絵を描く方法は様々であるためすべてがこの通りではない。しかし、初心者にとってはこの流れを通じて絵を描く技術を習得するのが望ましい。また、「線」と「塗り」の工程においては、使うことが想定される機能が異なる。したがって、二つの工程において独立した機能を推薦することができ、初心者が各機能を使

われる作業に紐づけることができる。以上の理由により、「線を引く作業」と「色を塗る作業」を分類することに重点を置いた。本論文では、デジタルで絵を描くという目的と、対象とするユーザを初心者に絞り、詳細な操作ログを取得する。操作ログに基づいてユーザが何をしているかを推定し、その人に合った機能を正確に推薦する。

本稿では次の構成をとる。まず、2 節で関連研究をまとめ、3 節で使用するアルゴリズムと提案手法について説明し、4 節で実験について説明し、5 節で提案手法の評価を行い、最後に 6 節でまとめを述べる。

2. 関連研究

本章では、グラフィックを扱うソフトウェアにおける機能推薦に関する研究を紹介する。

2.1. 大量のユーザデータを用いた研究

2009 年 Autodesk 社の Matejka らは、AutoCAD というソフトウェア製品にて、ユーザデータを取得し、そのデータに基づいて機能を推薦するシステムを提案した[1][2][3]。データセットは 16,000 人のユーザから集めた 4,000 万の{ユーザ, コマンド, 時間}からなるタプルである。ユーザからデータを集め、推薦アルゴリズムを適用して、各ユーザに特有の機能推薦を行う。推薦方法には、ユーザベースとアイテムベースの 2 つの手法がある。何れの手法も主な流れは、機能ベクトルを定義し、機能ベクトルに近いユーザまたはコマンドを抽出し、当該ユーザに推薦する。しかし、ユーザベース、アイテムベースというアプローチは、過去のコマンド履歴に基づいて推薦を行っており、現在行っている作業と被推薦対象となる機能を関連付けた推薦ができない。このため、機能について推薦し当該ユーザが理解したとしても、活用する場面が分からないという問題が生じる。

2.2. ウェブ文書を用いた研究

2015 年に Manitoba 大学の Khan らは、フリーソフトの GIMP を対象に、ウェブの検索結果をもとに「機能」と「作業」をマッピングし、そのマップを元に機能を推薦するシステムを提案した[4]。このマッピングは、QF-Graph[5]を用いて行われている。具体的には、GIMP で行った作業名が Google Suggest API に送られ[6]、検索結果（例えば GIMP how to と入力したときに、候補として出てくる作業）が推薦される。検索結果により、機能と作業の関連性を 0~1 の重みで表現すると共に、ユーザの機能の利用履歴から、作業と機能の関連度の大きい作業が選択される。しかし、初心者がソフトウェアを扱う場合、当該作業に必要な最低限の機能しか使わないことがあるため、コマンドだけで作業を判定

² <https://krita.org/jp/>

することができない場合がある。

2.3. まとめ

Autodesk 社の研究[1]では、作業と機能を関連付けた推薦ができないため、機能について知っていても使う場面が分からないという問題が生じる。ウェブ文書を用いた研究[4]では作業と機能をマッピングすることにより、その問題点を解決している。しかし、作業の推定に用いるのはコマンドだけであるため、正確に作業を推定することが難しい。そこでこれらの問題を解決すべく、本研究では、絵を描くという目的と、対象とするユーザをペイントソフト初心者に絞り、詳細な操作ログを取得することでより正確にユーザが必要とする機能を推薦することを目指す。

3. 提案手法

以下、3.1 で提案の概要、3.2 で操作ログについての説明、3.3 で分類した状態ごとに推薦する機能についての説明、3.4 では分類方法について述べる。

3.1. 提案概要

提案する手法の概要を図 3.1 に示す。ペイントソフトはオープンソースのフリーソフトである Krita を用いる。操作ログを取得できるように実装した Krita を使用してユーザに絵を描いてもらい、操作ログを取得する。取得した操作ログを SVM によって、ユーザが線を描く作業をしていたのか、塗る作業をしていたのか分類する。

絵を描くことにおいて、もっとも基本となる作業は、「線を描く」と「色を塗る」ことである。また、「線」と「塗り」の工程においては、使うことが想定される機能が異なるため、二つの工程において独立した機能を推薦することができる。具体的には、使っているブラシの種類と色、レイヤーが一致しているストロークのまとまりに対して、線であるか塗りであるかを分類する。このストロークのまとまりを以下、「ストロークセット」とする。これにより、操作ログにおいて当該部分が線を描いている部分なのか色を塗っている部分なのかを判定する。次に、その分類に基づき、線を描く作業、塗る作業を行っている過程それぞれに対してユーザが使用した機能(例:変形,透明度保護)を記録する。線を描く作業と色を塗る作業の各々について、あらかじめ推薦する機能を決めておき、ユーザが作業において使用していなかった機能を推薦する。この推薦は絵を描き終わった後に行われる。

3.2. 操作ログ

提案手法では、操作ログを取得しこれに基づいて機能を推薦する。操作ログとして記録される情報は、以下の 11 種類である。操作を行った時に付随して取得できる情報も示す。

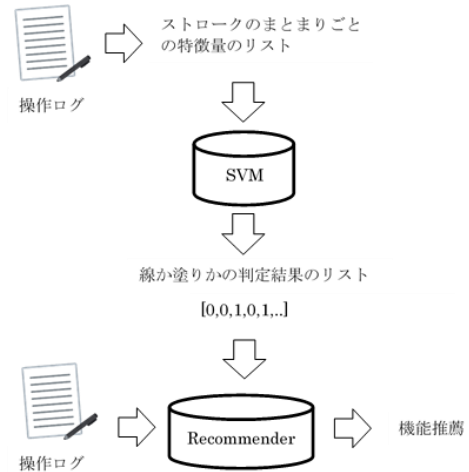


図 3.1 提案手法の概要

- ストローク
時間、消しゴムモードの有無、ブラシの種類、レイヤーの名前、レイヤーの ID、レイヤーの種類、レイヤーの合成モード、レイヤーの透明度、透明度保護の有無、透明度継承の有無、親レイヤーの名前、子レイヤーの名前、色、ブラシのサイズ、ブラシの透明度、ストロークの座標、ストロークの筆圧
- 元に戻す
- やり直す
- レイヤーを追加
レイヤーの種類
- バケツ
時間、消しゴムモードの有無、ブラシの種類、レイヤーの名前、レイヤーの ID、レイヤーの種類、レイヤーの合成モード、レイヤーの透明度、透明度保護の有無、透明度継承の有無、親レイヤーの名前、子レイヤーの名前、色、ブラシのサイズ、ブラシの透明度
- グラデーション
時間、消しゴムモードの有無、ブラシの種類、レイヤーの名前、レイヤーの ID、レイヤーの種類、レイヤーの合成モード、レイヤーの透明度、透明度保護の有無、透明度継承の有無、親レイヤーの名前、子レイヤーの名前、色、ブラシのサイズ、ブラシの透明度
- 選択
選択の種類
- 移動
- 変形
- 透明度変更
変更前の透明度→変更後の透明度
- レイヤーの合成モード変更
変更前の合成モード→変更後の合成モード

3.3. 分類方法

提案する手法は、推薦の前処理として取得した操作ログに対して以下の 3 つの処理を行う。

1. 操作ログの分割
2. 特徴量の抽出
3. SVM による線・塗りの判別

3.3.1. 操作ログの分割

ユーザが絵を描き終え、取得した操作ログに対して

以下の3点が一致し、連続しているストロークセットに分割する。

1. ブラシの種類
2. 色
3. 描きこんでいるレイヤー

絵を描く作業では、ブラシと色とレイヤーを選択して線を引くまたは色を塗る作業の繰り返しとなる。したがって、ブラシ、色、レイヤーを変更した瞬間がある作業の切れ目であると考えられるので、この分割を行う。具体的には、3.2で示した11種類のコマンドのうち、ストロークのみに着目して操作ログを分割する。ストロークの間にストローク以外のコマンドが挟まった場合は、元に戻すとやり直すも含め、無視して次のストロークの内容を見ていく。ストロークは連続しているもののみを対象とする。途中で別のストロークが挟まり、再び3点が一致しているストロークが登場する場合、それは別のストロークセットとして扱う。ストロークセットの合間に機能が使われた場合は、線の作業で使われる機能は線の作業で行われたものとし、色を塗る作業で使われる機能は色を塗る作業で使われたとする。ストローク以外のコマンドが挟まった後に3点が一致しているストロークが存在する場合は、同じストロークセットとして扱う。3点が異なるストロークが挟まった後に3点が一致しているストロークが存在する場合は、別のストロークセットとして扱う。

3.3.2. 特徴量の抽出

ストロークセットごとに、SVMに適用するための特徴量を抽出する。特徴量は以下の2点である。

1. ブラシのサイズ
2. 描画比

一般的に、線を引く作業は、細いブラシで、色を塗る作業は太いブラシで行われる。しかし、色を塗る作業で、細かいところを細いブラシで塗ることがある。色を塗るとき、一定の領域をまとめて塗っていることに着目し、その特徴量を取得できるように二つ目の描画比を取り入れた。

ブラシのサイズ

ブラシのサイズは、ストロークセットで使われたブラシのサイズの平均である。ストロークセットのうち、 i 番目のストロークのブラシのサイズを s_i 、ストロークセットのストロークの数を n とすると、一つ目の特徴量は式(1)で表せる。

$$\frac{\sum_{i=1}^n s_i}{n} \quad (1)$$

描画比

操作ログの座標とブラシのサイズと色から、ユーザがストロークセットで描いた画像を生成し、その画像

から描画比を取得する。描画比は、一定の領域がどのくらい塗られているかを示す。描画比は以下の4つの過程から生成された画像を元に抽出される。ここで扱う単位は画像のサイズを表すピクセル単位である。ストロークは座標の集まりで表現されるので、座標一つ一つに対してブラシのサイズの円をキャンバスにプロットすることで、キャンバスにそのストロークの描画を再現する。

1. ペイントソフトを起動し、最初に絵を描く領域のピクセル数を決める。これをキャンバスサイズと呼ぶ。ユーザが決定したキャンバスサイズと同じサイズの画像を生成する。本研究では一律 1000×1000 ピクセルで統一した。画像の色はストロークセットで使われている色以外の色とする。

2. 操作ログのブラシのサイズからプロットする円の半径 r を以下の手順で決める。

ブラシごとに形状と描かれ方が異なるが、ここでは一律円として扱った。また、ブラシごとに筆圧の影響による描画範囲の変化が異なる。筆圧の影響の定義の方法は、Kritaで各ブラシを使用したときに、表示されるブラシの描画範囲が縮む様子を目視で確認して定義した。プロットする円の半径を r 、ブラシごとの重みを w 、ブラシのサイズを s とすると、円の半径は式(2)である。Sketchシリーズのブラシは、ブラシのサイズに依存した描画ではなかったため、 $r=5$ の固定値とした。

$$r = s * w * 0.5 \begin{cases} 0 < w < 1 \\ 0 < s < 1000 \end{cases} \quad (2)$$

3. ストロークの座標と2で求めた円の半径から、1で生成した画像に座標一つに対して一つの円をプロットしていき、すべての座標に対してプロットすることにより、ストロークを描画する。

4. 3をストロークセット内のストロークに対して行うことにより、ストロークセットを描画する。

図3.2は一つのストロークセットから上記に示した手順で生成した画像のイメージである。図3.2を用いて2つ目の特徴量である描画比について説明する。a, b, v, hは以下のように定義する。

- a: 描画されたピクセルが、最も連続している長さ
- b: 描画された部分の最小の長方形の高さまたは幅
- v: 垂直方向
- h: 水平方向

画像を列と行から成るピクセルの行列とすると、一列、一行ずつピクセルを見ていく。一列、一行の中でストロークセットの色が連続しているもので、最も長いものをその列、行の長さとする。列方向、行方向でそれぞれ最も長いもの a を選択する。 a を描かれている部分の最小の長方形の長さ b で割ったものが描画比である。

最終的に使用する特徴量を描画比 R とすると、R は式(3)で定義される。

$$R = \max\left(\frac{a_v}{b_v}, \frac{a_h}{b_h}\right) \quad (3)$$

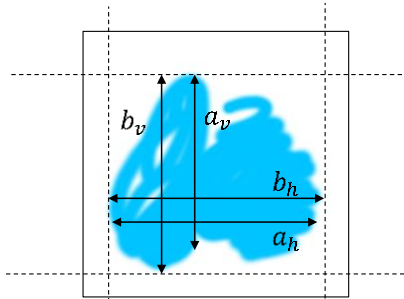


図 3.2 操作ログから生成した画像から特徴量の取得

3.3.3. SVM による線・塗りの判別

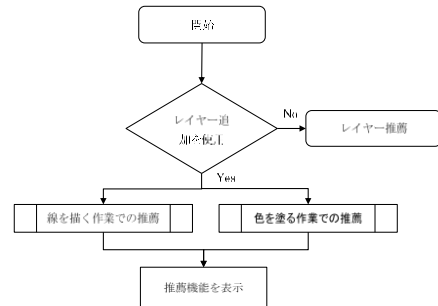
SVM は教師あり学習を行うパターン認識モデルの一つである。学習データにより構築されたモデルを用いて、未知のデータを推測して分類することができる。3.3.1 で分割したストロークセットにおいて、3.3.2 で抽出した特徴量のリストが入力である。出力は、線か塗りかの分類を示す 0,1 のリストである。SVM のカーネル関数は、RBF カーネルを用いた。RBF カーネルを用いるにあたって、コスト C とガンマ γ の 2 つのパラメータを設定する必要がある。4.1 で行ったモデル構築のための実験で得たデータを用いてグリッドサーチを行った。被験者ごとに Leave-one-out とグリッドサーチを用いて、 $C=32768.0$ 、 $\gamma=0.401705613283$ のとき正解率が最も高くなった。推定正解率は 88.4% である。これらの値を評価実験で用いる SVM のモデルのパラメータとして用いる。

3.4. 分類に基づいた機能推薦

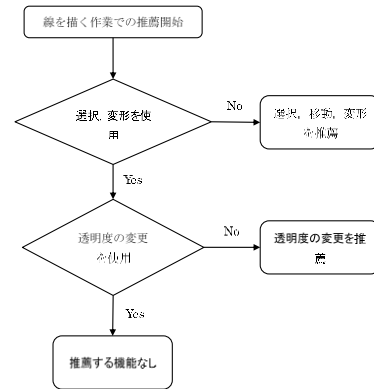
分類に基づいた機能推薦では、予め推薦候補となる機能の中からユーザーが利用しなかった機能を推薦する。ここで、線と塗りの工程において推薦する機能は、あらかじめ決定しておく。それぞれの機能は使われる作業が想定されているため、操作ログによってユーザーが行っている作業を判別することができれば、その作業に有効な機能が推薦できる。その作業を行うのに有効な機能であるにも関わらず、ユーザーが使っていないならば、ユーザーがその機能を知らない可能性が高い。

推薦方法は、ユーザーが「線を描く作業」を行っているとき、「色を塗る作業」をしているときそれぞれにおいて、ユーザーが使用しなかった機能を推薦するという方法である。分類方法は、「ストロークセットを線か塗りか分類する」というものである。ストロークセットへの分割は、ストロークの内容だけを見て、他 10 種類

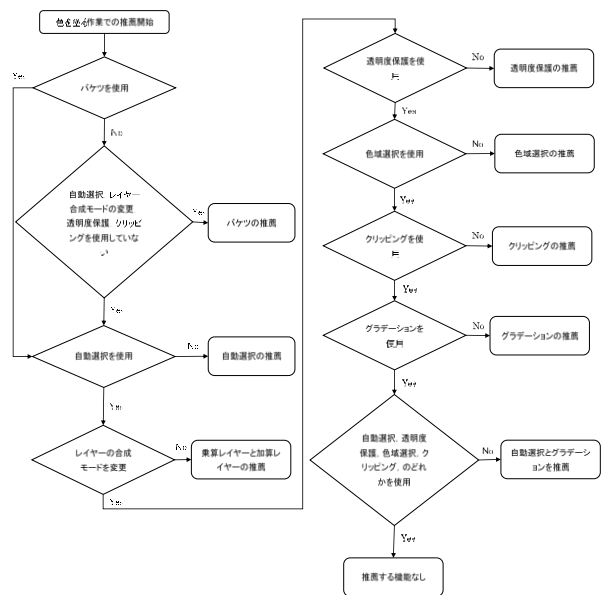
の機能は無視する。したがって、線を引く作業のストロークセットと色を塗る作業のストロークセットの切れ目で使われた機能は、どちらの作業で使われたのか判別できない。その場合は線の作業で使われる機能は線の作業で行われたものとし、色を塗る作業で使われる機能は色を塗る作業で使われたとする。複数機能を推薦できる場合はあらかじめ決めておいたフローチャート(図 3.3)に基づいて推薦を行う。



(a) フローチャートの全体図



(b) 線を描く作業での推薦のフローチャート



(c) 色を塗る作業での推薦のフローチャート

図 3.3 機能推薦のフローチャート

4. モデル構築のための実験

本研究の実験では、被験者実験を二回行った。一回目の被験者実験は、SVMのモデルを構築するための予備実験であり、二回目の被験者実験は推薦システムの評価のための実験である。

一回目の被験者実験では、被験者はペイントソフト初心者、ペイントソフト上級者の被験者のどちらにも実験に参加してもらった。二回目の被験者実験は、本研究で提案する推薦システムの評価を行うためのものである。そのため、被験者はペイントソフト初心者に限定した。なお、初心者と上級者は、一回目の実験でのアンケートに基づいて定義した。

ペイントソフト初心者

以下の2つの条件のどちらかを満たす人

1. ペイントソフトの使用年数が一年未満
2. ペイントソフトを用いて一から描いた絵の枚数が10枚以下である

ペイントソフト上級者

以下の2つの条件の両方を満たす人

1. ペイントソフトの使用年数が一年以上
2. ペイントソフトを用いて一から描いた絵の枚数が11枚以上である。

ペイントソフトは Krita(バージョン 3.0)を操作ログが取れるように実装しビルドしたものを使用する。ペンタブレットは wacom 社の intuosCTH-680 を使用する。

4.1. 実験概要

この被験者実験は、SVMのモデルを構築するための予備実験である。4.1で実験概要について、4.2で収集したデータについて述べる。

一回目の実験は以下の8手順で行った。

1. 実験担当者がペンタブレットをPCに接続する。
2. 実験担当者がペイントソフトを起動する。
3. 被験者にペンタブレットとペイントソフトの使い方を説明する。
4. 5分間、ペイントソフトに慣れるため、ペンタブを使ってペイントソフトを操作してもらう。
5. 操作ログを取得できるモードでペイントソフトを起動し直す。
6. 被験者が50分以内に指定された絵を描く。
7. 実験担当者がペイントソフトを終了する。
8. 被験者にアンケートに答えてもらう。

被験者がペイントソフト初心者であった場合、被験者のレベルに合わせて、50分で描き上げることができそうな絵を指定する。被験者がペイントソフトに精通している人であった場合、50分で好きな絵を描いてもらった。絵によって特徴量が偏ってしまわないように、複数の絵を用意して被験者に描いてもらう絵がばらけるよう指定する。指定する絵は機能を推薦できるよ

にするために以下の特徴を持たせた。

1. 輪郭線がある。
2. 影になっている部分がある。
3. 光が当たっている部分がある。
4. 描くオブジェクト自身の色は2色以上で構成される。

4.2. 収集データ

被験者は18歳~25歳までの男性9人女性20人で、計29人である。ペイントソフト初心者は18人、ペイントソフト上級者は11人である。被験者1人に対して、1~2枚の絵のデータと9個から253個のストロークセットのデータを得た。操作ログがストロークセットに分割される個数は個人差があるが、上級者の方が多くなる傾向にあった。結果、SVMでの学習に用いる収集データとして、1,974個のストロークセットのデータを得られた。学習データのラベル付けは、生成した画像を目視で確認し、線か塗りかのラベルを手動で設定した。

5. 推薦システムの評価

本節では、提案した推薦システムの評価を行う。二回目の被験者実験では、被験者に推薦された機能についてのアンケートを取り、推薦システムを評価する。

5.1で二回目の被験者実験による評価実験について説明し、5.2で評価指標について説明し、5.3で実験結果に基づいた評価を行う。

5.1. 評価実験

本被験者実験は推薦システムの評価のための実験である。ペイントソフト初心者である被験者に絵を描いてもらい、機能を推薦する。推薦された機能に対して、被験者にアンケートを取り、推薦システムを評価する。実験は以下の10手順で行う。

1. 実験担当者がペンタブレットをPCに接続する。
2. 実験担当者がペイントソフトを起動する。
3. 被験者にペンタブレットとペイントソフトの使い方を説明する。
4. ペイントソフトに慣れるため、5分間ペンタブを使ってペイントソフトを操作してもらう。
5. 操作ログを取得できるモードでペイントソフトを起動し直す。
6. 被験者が50分以内に指定された絵を描く。
7. 実験担当者が推薦システムを起動する。
8. システムが機能を推薦する。
9. 被験者に推薦内容についてのアンケートに答えてもらう。
10. 実験担当者がペイントソフトを終了する。

被験者には図5.1で示す絵を描いてもらう。被験者が絵を描き終えた後、取得した操作ログを推薦システムに入力し、機能を推薦する。被験者に推薦された機能についてのアンケートを取り、推薦の評価をする。

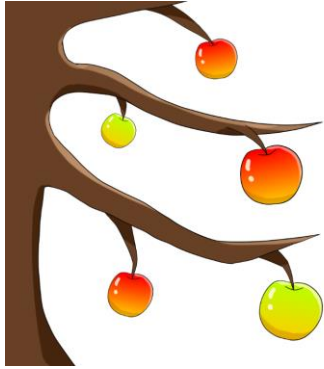


図 5.1 評価実験で被験者に描いてもらう絵

5.2. 評価指標

評価指標として、[4]で用いられている図 5.2 の指標を用いる。図 5.2 の指標を用いるため、図 5.3 のアンケートにより評価を行う。図 5.3 のアンケートは推薦された機能ごとに回答してもらう。

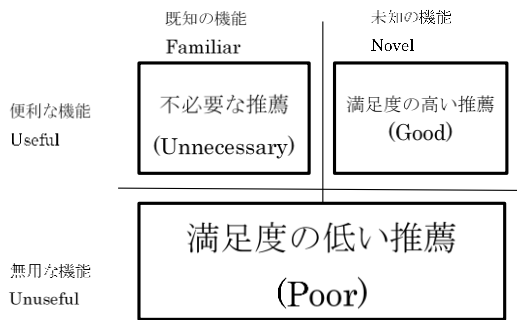


図 5.2 評価指標
[1]を元に作成

問 1.	この機能を知っていましたか？
1.	よく知っていた。
2.	どちらかといえば知っていた。
3.	どちらとも言えない。
4.	どちらかといえば知らなかった。
5.	全く知らなかった。
問 2.	この機能を今後も使いたいと思いますか？
1.	とてもそう思う。
2.	どちらかといえばそう思う。
3.	どちらとも言えない。
4.	どちらかといえばそう思わない。
5.	全くそう思わない。

図 5.3 推薦された機能に関するアンケート

5.3. 推薦の評価

被験者は 19 歳～34 歳までの男性 11 人女性 5 人で、計 16 人である。5.3.1 で SVM によるストロークセットの「線」と「塗り」の判別について、5.3.2 でアンケ

ートによる推薦機能についての評価を行う。

5.3.1. SVM によるストロークセットの「線」「塗り」判別評価

被験者 1 人に対して、1 枚の絵のデータと 8 個～50 個のストロークセットのデータを得た。結果、SVM での学習に用いる収集データとして、282 個のストロークセットのデータを得られた。学習データの正解ラベル付けは、モデル構築のための実験と同じく、生成した画像を目視で確認し、線か塗りかのラベルを手動で設定した。SVM は RBF カーネルを用いた。パラメータはモデル構築のための実験で集めたデータを元に、Leave-one-out とグリッドサーチを用いて求めた $C=32768.0$ 、 $\gamma=0.401705613283$ を用いる。表 5.1 に被験者ごとのデータ数と正解率を示す。

表 5.1 を見ると、12 人の被験者が 75% を超える正解率なのに対し、4 人の被験者が 55% の正解率を下回った。ストロークセットから生成した画像と実験中の観察を元に考察したところ、パケツツールのみ用いて色塗りをを行っている被験者に見られる傾向であったパケツツールは線で囲まれた部分を塗りつぶす機能である。色を塗りたい部分を線で囲ってパケツツールを用いて色を塗り、ブラシでは色を塗らない場合、本研究の手法ではストロークのデータは実質線を引く作業しか行っていないことになってしまう。比較的塗りだと判定しやすい広範囲の塗りはパケツで行われてしまうため、ストロークセットでの描画範囲は線と塗りの判別がしにくいような細かい描画となっていると考えられる。

表 5.1 評価実験における被験者ごとのデータ数と正解率

被験者	ストロークセット数	正解率[%]
1	12	33.33
2	8	37.50
3	11	45.45
4	11	54.55
5	14	78.57
6	10	80.00
7	50	80.00
8	21	80.95
9	18	83.33
10	25	84.00
11	20	90.00
12	14	92.86
13	16	93.75
14	19	94.74
15	24	100.00
16	9	100.00
平均正解率		76.38

5.3.2. アンケートによる推薦された機能の評価

推薦された機能に対して、図 5.11 のアンケートを推薦された機能ごとに被験者に回答してもらった。表 5.2 と図 5.17 に満足度の高い推薦、不必要な推薦、満足度

の低い推薦の内訳を示す。満足度の高い推薦は 60%であった。

評価指標を参考にした研究[4]では、問1の選択肢の1~3を既知の推薦、4,5を未知の機能と定義し、問2の選択肢の1,2を便利な機能、3~5を無用な機能であると定義しており、本評価でも同様の基準を採用した。

- 問1に対して、機能を知っていた場合でも2の選択肢「どちらかと言えば知っていた」と回答する人が多かった。機能自体はなんとなく知っていても使い方や使いどころが分からないためであると考えられる。機能のことを知っていれば調べて詳しく知ることができるが、使いどころを含めて推薦すれば知っていた状態でも有効な推薦となる可能性がある。
- 問2に対して、2の選択肢:「どちらかと言えばそう思う」を選択したものが多かった。本実験では、対象者をペイントソフト初心者に限定しているため、使っていない機能を推薦すると、上級者に比べて便利だと思う傾向がある。図5.12の記述式アンケートと絵を描いている最中の様子を観察したところ、絵を描いている時に困っていたことに対して、推薦された機能によって解決できるものだった場合、問2の1の選択肢:「とてもそう思う」を選ぶ傾向にあるように考えられる。

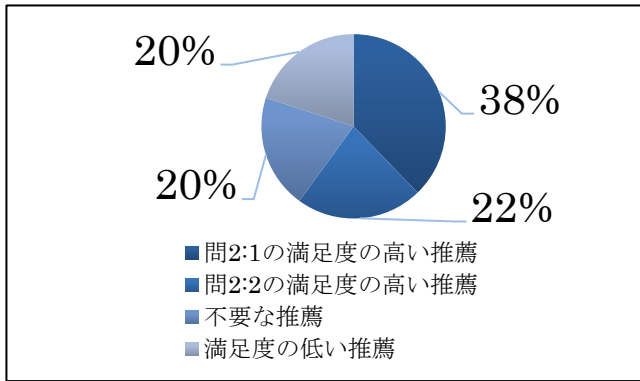


図 5.3 評価の内訳

6. おわりに

本稿では、ペイントソフトにおいて操作ログを取得し、初心者に対して機能を推薦する手法を提案した。初心者でも最低限用いるブラシの使い方だけを使用して、操作ログを「線を引く作業」と「色を塗る作業」の作業に分類し、その分類に基づきユーザが使用しなかった機能を推薦した。「線を引く作業」と「色を塗る作業」への分類を行うため、操作ログに記録されているストロークをブラシの種類と、色と、レイヤーの3点が一致しているストロークセットに分割し、ストロ

ークセットを SVM により「線を引く作業」と「色を塗る作業」に分類した。その分類に基づき、「線を引く作業」と「色を塗る作業」においてユーザが使った機能を記録し、使っていない機能を推薦した。被験者に推薦された機能に対するアンケートを取る評価実験の結果、ユーザが知らなかった機能でかつ便利だと感じる機能の推薦が 60%の割合で行われた。分類を「線を引く作業」と「色を塗る作業」だけでなく、様々な作業に拡張し、その作業に対応する機能へ拡張することが今後の課題である。

参考文献

- [1] Justin Matejka, Wei Li, Tovi Grossman, George Fitzmaurice, “CommunityCommands: command recommendations for software applications”, Proc. of ACM UIST, pp. 193-202, 2009.
- [2] Wei Li, Justin Matejka, Tovi Grossman, Joseph A. Konstan, George Fitzmaurice, “Design and evaluation of a command recommendation system for software applications”, Proc. of ACM TOCHI, Article No. 6, 2011.
- [3] Justin Matejka, Wei Li, Tovi Grossman, George Fitzmaurice, “Deploying community commands: a software command recommender system case study”. Proc. of AAAI ACM, pp. 2922-2929, 2014.
- [4] Luca Benedetti, Holger Winnemöller, Massimiliano Corsini, Roberto Scopigno, “Exploring Personalized Command Recommendations based on Information Found in Web Documentation”. Proc of ACM IUI, pp. 225-235, 2015.
- [5] Adam Fourney, Richard Mann, Michael Terry, “Query-feature graphs: bridging user vocabulary and system functionality”, Proc. of ACM UIST, pp. 207-216, 2011
- [6] Adam Fourney, Richard Mann, Michael Terry, “Characterizing the usability of interactive applications through query log analysis”, Proc. of ACM CHI, pp.1817-1826, 2011