

## Web上で編集/派生可能なイラストツールの研究

神原 啓介<sup>†1</sup> 永田 周一<sup>†2</sup> 塚田 浩二<sup>†1,†3</sup>

Web上では人々によって描かれた多くの絵が公開・共有されているが、それらを再利用することは難しく、あまり行われていない。もし他の人の描いた絵を簡単に再利用でき、また再利用しやすい形で公開することができれば、より多くの人々が手軽に絵を活用できるようになると考えられる。そこで我々はWebブラウザ上で絵を編集でき、他の人の描いた絵から「派生」させて新しい絵を作成することのできる2種類のWebアプリケーション:「Willustrator」と「TwitPaint」を開発した。ドロー系であるWillustratorは絵を再編集/再利用しやすいという特徴を持つ。一方、ペイント系のTwitPaintはTwitterと連携した絵によるコミュニケーションを特徴とする。両ツールを長期運用し、得られた派生データや事例、知見などをもとに、相違や特徴、問題点を分析した。さらに今後のWeb上で編集/派生可能なイラストツールの課題や将来の可能性についても検討を行った。

### A Study of Illustration Tools for Image Editing and Derivation on the Web

KEISUKE KAMBARA,<sup>†1</sup> SHUICHI NAGATA<sup>†2</sup>  
and KOJI TSUKADA<sup>†1,†3</sup>

Although many illustrations are shared on the Web, most people feel difficulty to reuse them. To solve this problem, we propose a social illustration system which supports people draw illustrations in reusable format and share them in social web sites. Based on the above concept, we developed two Web applications, Willustrator and TwitPaint, which allow people to draw images derived from other illustrations and edit them easily on common Web browsers. Willustrator is a “drawing” tool which focuses on drawing illustrations in reusable format. TwitPaint is a “painting” tool which allows people to share their illustrations easily in Twitter. We have put these applications into practice for a long period, and examined various aspects based on the experiences. Finally, we discuss the future work of the social illustration system.

### 1. はじめに

Web上では人々によって描かれた多くの絵やクリップアート、図、イラストが公開/利用されている。なかでもイラストの共有に特化したコミュニティサイトは昔から数多く運営されており、海外では deviantART<sup>\*1</sup>、日本国内では Pixiv<sup>\*2</sup>といったサイトが特に人気を博している。このようなイラスト投稿サイトはアーティストやイラストレータ、漫画家といった専門家や絵を描くのが得意な人(熟達者)を中心に利用されているが、絵を描くスキルや専門知識をあまり持たない人(非熟達者)によるイラストの投稿や共有はあまり行われていない。

一方、非熟達者であっても仕事や趣味、コミュニケーションの中で絵を活用する機会は多い。ワープロの文書やブログの記事、プレゼンテーション用のスライド、ポスタやパンフレットを作る際には絵が頻繁に用いられる。そして Adobe Photoshop や Illustrator, Visio<sup>\*3</sup>, OmniGraffle<sup>\*4</sup>のようなグラフィックスソフトウェアや、Microsoft Word や PowerPoint のようなドキュメント作成ソフトウェアの普及によって、非熟達者でも絵を利用することは珍しくなくなった。

しかし、このようなデスクトップアプリケーションで描かれた絵は複数人での協同編集や、他の人の描いた絵を再利用するといったことが難しく、さらに自分の描いた絵を再利用しやすい形で共有することも面倒である。ブログや SNS, Wiki などの Web アプリケーションによって情報共有やコミュニケーションの敷居は下がったものの、Web上での「非熟達者による絵の共有」や「絵の協同編集, 再利用」はまだ発展途上といえる。

デスクトップアプリケーションを使って描かれた絵が共有されにくい理由の1つに、ローカルPCからWeb上へのファイルのアップロード作業が面倒なことがあげられる。テキスト情報であればブログや Wiki などを利用してブラウザ上ですぐに編集し直すことができる

†1 お茶の水女子大学お茶大アカデミックプロダクション  
Academic Production, Ochanomizu University

†2 慶應義塾大学大学院政策・メディア研究科  
Graduate School of Media and Governance, Keio University

†3 科学技術振興機構さきがけ  
JST PRESTO

\*1 <http://www.deviantart.com/>

\*2 <http://www.pixiv.net/>

\*3 <http://office.microsoft.com/ja-jp/visio/>

\*4 <http://www.omnigroup.com/applications/omnigraffle/>

が、絵は編集し直すたびにファイルをアップロードする必要がある。また、ファイル名をつけて分類保存するといったファイル管理の手間も、ブラウザ上でのテキスト編集に比べて面倒な理由の 1 つである。

もし、Web 上で公開/共有されている絵を手軽に再利用でき、さらに他の人が再利用しやすい形で簡単に共有することができれば、より便利で手軽に絵を活用できると考えられる。

そこで我々は「Web ブラウザ上で絵を直接編集でき、他の人の絵をもとに再編集・再利用を可能にする」2 種類の Web アプリケーションを開発した。1 つはドロー系の「Willustrator」で、もう 1 つはペイント系でかつ Twitter との連携を特徴とする「TwitPaint」である。この両ツールを一般に公開し長期運用することで、Web 上での絵の作成や再利用に関するデータや知見をまとめ、今後の課題や将来の可能性についても検討を行う。

### 1.1 絵の派生

Willustrator と TwitPaint はともに「他の絵から派生させて新しい絵を描ける」ことが大きな特徴となっている。ここでは以下の 2 つの条件を満たすことを「絵の派生」とする。

- ある絵を複製後、再編集して、元の絵とは別の新しい絵を作成する。
- 再編集して作られた絵と、その元になった絵が相互にリンクされる。

1 つめの条件により、他の絵を再編集する際に上書きしてしまうのではなく、1 度複製してから編集することで、元の絵を残したままにすることができる。また 2 つめの条件により、再編集してできた絵と元になった絵が相互にリンクされることで、あとから元になった絵や、逆にその絵をもとにして描かれた絵をたどって見るようになる。

派生が次々と繰り返されることで、1 つの絵が様々に変化・分岐してゆき、複数の人がその一連の流れに関わることになる。本研究ではこの「絵の派生」を特徴とする両ツールを実際に運用することで、絵の変化・分岐の課程を楽しむという「新しい絵の楽しみ方」や、絵の派生を活用した「新しい表現の可能性」、絵と絵・作者と作者がつながることによる「新しいコミュニケーションの発生」といった、様々な可能性を探った。

## 2. Willustrator

Willustrator とは Web ブラウザ上で使えるドローツールである(図 1)。絵のベクタデータを Web 上で保存・共有することで、他の人の描いた絵を手軽に再利用して新しい絵を描くことができる。また、ドロー系ツールであるため、ペイント系ツールに比べて再編集しやすいことが特徴となっている。Willustrator は現在 <http://willustrator.org/> で運用している。

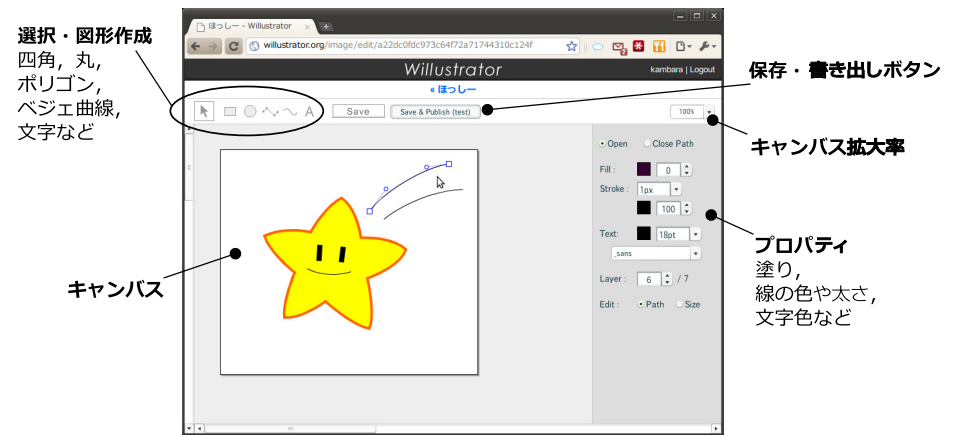


図 1 Willustrator の画像編集画面

Fig. 1 Image editing screen of Willustrator.

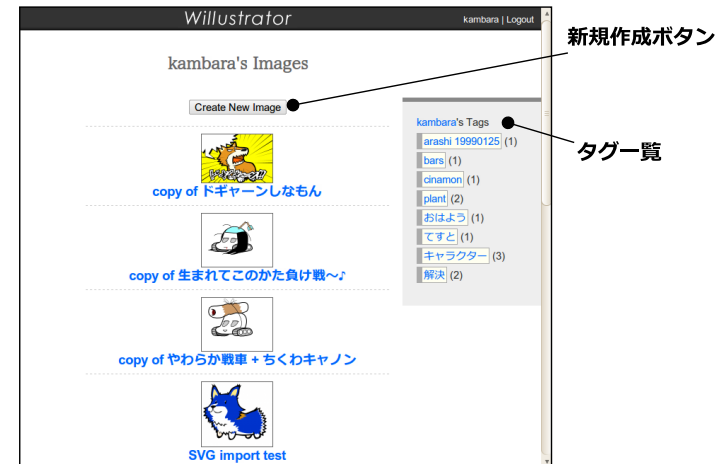


図 2 Willustrator のユーザ個人ページ

Fig. 2 A user's page in Willustrator.

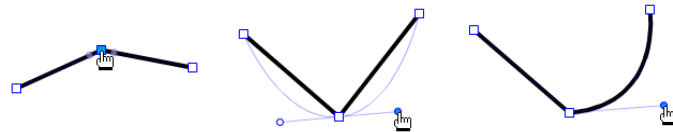


図 3 ベジエ曲線の編集 (Willustrator)  
Fig. 3 Editing bezier curve (Willustrator).

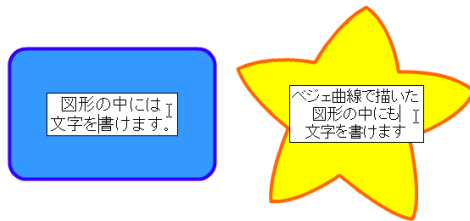


図 4 図形内テキスト (Willustrator)  
Fig. 4 A text in a shape (Willustrator).

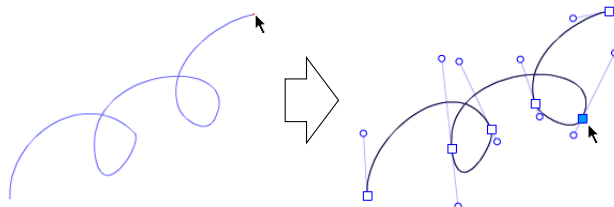


図 5 フリーハンドツール (Willustrator)  
Fig. 5 Freehand tool (Willustrator).

## 2.1 機能と使い方

Willustrator の描画機能やサイトの利用方法, 再利用のための機能について述べる.

### 2.1.1 画像の一覧と新規作成

Willustrator にログインすると, そのユーザの個人ページ (図 2) が表示され, 自分の描いた画像を一覧できるほか, 新規作成ボタンを押すと新しい絵を描ける.

### 2.1.2 描画機能

Willustrator の画像編集画面は図 1 のようになっている. 丸や四角, ポリゴン, テキストといった基本的な図形のほかに, 図 3 のようにベジエ曲線を編集することで複雑な図形

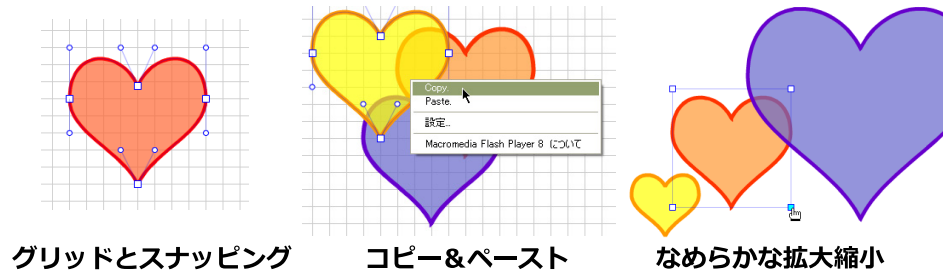


図 6 基本的な図形編集機能 (Willustrator)  
Fig. 6 Shape editing functions (Willustrator).



図 7 派生元および派生先の絵へのリンクと派生ボタン (Willustrator)  
Fig. 7 A source image, a derived image and a derivation button (Willustrator).

も描くことができ, 図 4 のようにすべての図形の中にはテキストを書ける. 図 5 のようにフリーハンドツールを使うことで, ベジエ曲線に不慣れなユーザでも簡単にきれいなベジエ曲線を描けるようになっている.

また図 6 のようなグリッド・スナッピング, コピー&ペースト, 拡大縮小といった, ドロー

ツールとしての基本的な機能を備えている。

### 2.1.3 絵の派生による再利用

Willustrator の特徴の 1 つは「他の絵から派生させることで再利用できる」ことである。たとえば図 7 上の絵をもとにして描きたい場合、派生ボタンを押すとその絵が複製され、編集することができる。また、派生元になった絵や、派生によってできた絵をリンクでたどって見ることができる(図 7)。

## 2.2 実装

Willustrator の Web アプリケーション(サーバ)は Ruby on Rails で実装し、ユーザ認証にははてな認証 API<sup>\*1</sup>を用いた。

画像編集機能は Adobe Flash で実装した。ベクタ画像の保存形式は独自の XML になっている。このベクタ画像の PNG 形式へのラスタライズはクライアント側(Flash)で行い、生成した PNG 画像をサーバへ送信・保存している。

## 3. TwitPaint

TwitPaint とは、Web ブラウザ上で絵を描けるペイントツールであり、描いた絵を自動的に Twitter に投稿することで、絵を通じてコミュニケーションすることができる。さらに、他の絵から派生させたのち再編集する(TwitPaint 内では Remix と呼ぶ)ことができ、この派生を活かすことで、複数のユーザが参加する絵の作成や、絵を使った遊び、Twitter にはないコミュニケーションを可能にする。

TwitPaint は現在 <http://twitpaint.com/> で運用している。

### 3.1 機能と使い方

TwitPaint の描画機能や Twitter との連携や派生機能について述べる。

#### 3.1.1 絵の描画と Twitter への投稿

TwitPaint には Twitter のアカウントでログインすることができ、ログインするとすぐに描き始められる。描画機能はシンプルで、ペイントブラシの太さと色、透明度を設定できるほか、アンドゥ機能を備えている(図 8)。

また、1 つの絵につき最大 140 ストロークだけ描けるようになっており、図 8 のようにキャンパスの右上には残りストローク数を表すカウンタが表示されている。140 ストロークという制限は、Twitter が 1 投稿につき 140 文字に制限することで短く気軽な投稿を促し



図 8 TwitPaint の画像編集およびコメント入力画面

Fig. 8 Image editing and text comment screen of TwitPaint.



図 9 TwitPaint の派生機能

Fig. 9 Derived images and a source image of TwitPaint.

ていることになっている。

絵を描き終わったあと、コメントを書いて投稿ボタンを押すと画像が保存され、さらに画像ページの URL とコメントが自動的に Twitter へ投稿される。

\*1 <http://auth.hatena.ne.jp/>



図 10 みんなの作品 (TwitPaint)  
Fig. 10 Everyone's works (TwitPaint).

### 3.1.2 派生 (Remix) 機能

TwitPaint では、画像一覧画面などから他の人の描いた絵を選んで詳細表示するだけで、その画像の上に描きこむことができる (図 9)。そして他の絵に加筆・編集した後に投稿すると派生になる。派生元になった絵や、派生させてできた絵は図 9 の下部のように表示され、リンクをたどって見ることができる。

#### 3.1.3 みんなの作品

「みんなの作品」というページでは TwitPaint のユーザが描いた絵を見ることができ、新着順の他に以下の 3 通りのランキングをそれぞれ月別・全期間で表示する (図 10)。

- 話題賞：コメントの多い順
- 元ネタ賞：その絵から派生した絵 (子画像) が多い順
- 継続賞：派生の連鎖 (世代) が長い順



図 11 お題ページ (TwitPaint)  
Fig. 11 Requesting for images (TwitPaint).

### 3.1.4 お題

「お題」ページでは図 11 のように「あなたの大好物は?」「猫描いて!」といったお題を投稿することができ、他の TwitPaint ユーザがそのお題に従って絵を描いて投稿する。

### 3.2 実装

TwitPaint の Web アプリケーションは PHP で、ペイント部は Flash で実装した。ペイント部で描かれた絵 (ビットマップ) は Flash 側で PNG 形式に圧縮後、サーバに送信・保存される。派生させて描く場合はその PNG 画像を読み込みビットマップとして編集する。

## 4. 運用と結果

Willustrator は 2006 年 1 月から、TwitPaint は 2009 年 7 月からそれぞれ運用を開始した。2010 年 6 月時点での利用状況を表 1 に示す。また、図 12 のグラフは、各ユーザがそれぞれ何枚の画像を作成したかを示す。

表 1 の「派生ツリー」とは、ある初代画像 (親画像を持たない画像) から派生した子画像と、さらにそこから派生したすべての子孫画像をまとめて 1 つのツリーとして呼び表したものである。1 回の派生ごとに世代が増え、親/子を 2 世代、親/子/孫を 3 世代という順

表 1 Willustrator と TwitPaint の利用統計 (2010 年 6 月時点)  
Table 1 Usage statistics of Willustrator and TwitPaint.

	Willustrator	TwitPaint
登録ユーザ数 (人)	768	15,954
総画像数 (枚)	1,403	53,999
派生により作られた画像数 (枚)	326 (23%)	10,820 (20%)
派生ツリーの子孫画像数の平均 (枚)	2.54	2.19
派生ツリーの世代数の平均 (世代)	2.12	2.86
最大世代数 (世代)	5	84
子を持つ画像のうちの子の数の平均 (枚)	1.28	1.14

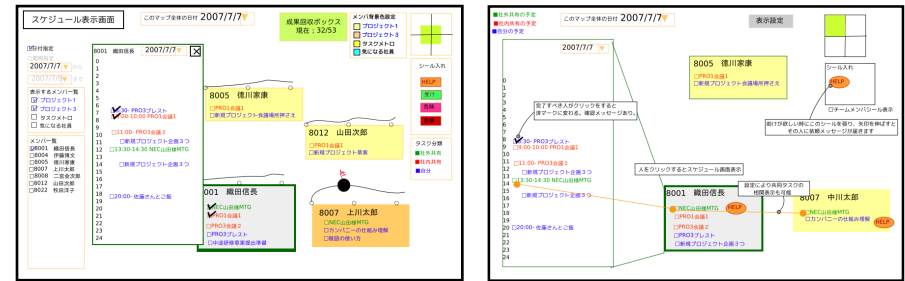


図 13 GUI のモックアップ作成 (Willustrator)  
Fig. 13 Creating mock-up images of GUI (Willustrator).

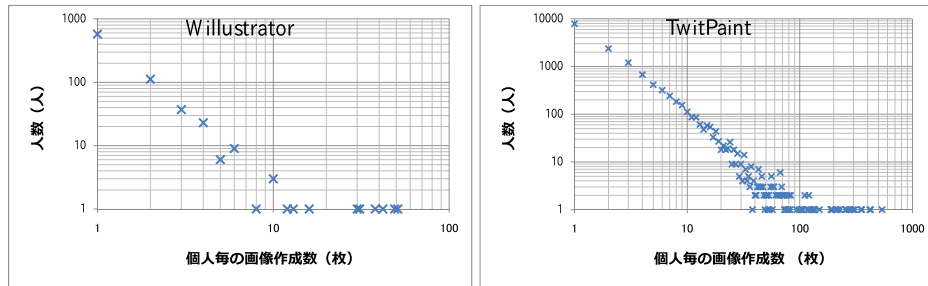


図 12 個人ごとの画像作成数とその人数の関係．各ユーザがそれぞれ何枚の画像を作成したかを表す  
Fig. 12 Relation between the number of images drawn by each user and the number of users.

に数える．また、表 1 の「派生ツリーの子孫画像数」および「派生ツリーの世代数」では 1 世代のみの画像、すなわちまったく派生されなかった画像はツリーとして数えていない．

#### 4.1 Willustrator の利用事例

Willustrator の特徴的な利用事例として「図解」「キャラクタと素材の合成」について述べる．

##### 4.1.1 図 解

Willustrator を用いて GUI (Graphical User Interface) のモックアップ作成や図解を作成した事例を紹介する．

図 13 では、ソフトウェアの GUI のモックアップ作成に使われた．この例では元となる画面から派生させることで 4 通りの画面を作成している．

図 14 ではブログ記事に載せるため図を作成している．さらに他の人がこの図から派生させて描くことで、図の問題点を指摘するような、新しい意味を持つ図を作成した．



図 14 ブログの記事用の図解とその派生 (Willustrator)  
Fig. 14 Figures for blog entries and the derivation (Willustrator).

図 15 では四象限の元となる図から派生させることで、具体的な四象限の図を作成している．この例では一種のテンプレートとして派生機能を利用している．

##### 4.1.2 キャラクタ

Willustrator でキャラクタを描いた例が多く見られた．

図 16 では、ある人が描いた白黒のキャラクタから別の人が派生させて、そのキャラクタに色を塗っている．このようにキャラクタから派生させて色を塗る、また新しいパーツを付け加えるといった使い方が見られた．

図 17 では、素材となる 2 つの絵 (犬と吹き出し文字) を組み合わせて新しい絵を作っている．さらにこの素材を描いたユーザは図 18 のように、一種の素材集として使えるようにキャラクタの画像をまとめて載せていた．

元となるキャラクタ画像は直接 Willustrator で描かれたものよりも、Adobe Illustrator などで描かれたものをインポートしているものが見られた．



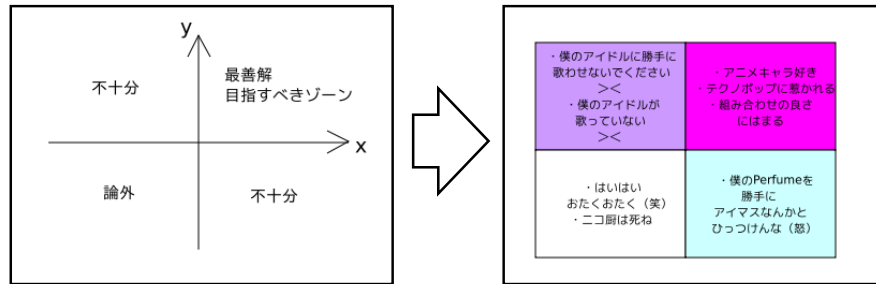


図 15 四象限図とその派生 (Willustrator)

Fig. 15 Four quadrant matrix and the derivation (Willustrator).

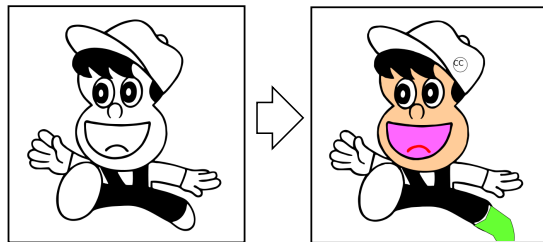


図 16 キャラクタに色を塗る (Willustrator)

Fig. 16 Coloring in a picture of a character (Willustrator).

## 4.2 TwitPaint の利用事例

TwitPaint は利用目的ごとに派生の仕方いくつかの傾向が見られた。

### 4.2.1 テンプレート

最も多くの子画像を持つ図 19 では、吹き出しの描かれた画像から派生させて、別のユーザが吹き出しの中に文字や絵などを書き足している。同様に吹き出しを書き込むようなテンプレートとなる画像を作ることで、多くの人が書き足して遊ぶ例が見られた。

### 4.2.2 ゲーム

派生ツリー内の子孫画像数が 2 番目に多かった図 20 では、派生を利用して三目並べをしている。元となる「井」形の画像から別のユーザが派生させて や x を書き込むことでゲームが進行している。通常の三目並べのように 2 人のユーザが対戦するのではなく、途中で枝分かれして派生させることで 1 つのツリーの中で複数人が対戦しているのも特徴的である。

図 21 では絵を使ったしりとりをしている。最初のユーザが「リンゴ」の絵を描いたあと、

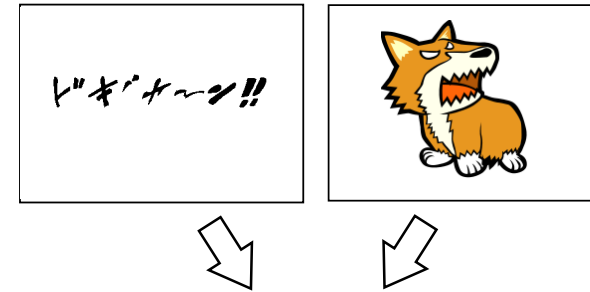


図 17 図形の合成 (Willustrator)

Fig. 17 Synthesis of figures (Willustrator).



図 18 キャラクタ素材 (Willustrator)

Fig. 18 Cliparts of characters (Willustrator).

別のユーザがそれから派生させて、しりとりとしてつながるように「ゴマ」「丸」というように次々と絵を足すことで派生が長く続いている。

### 4.2.3 力作 絵画

時間をかけて書き込むことで 1 つの絵を仕上げていく、力作といえる絵を描くユーザが見られた。

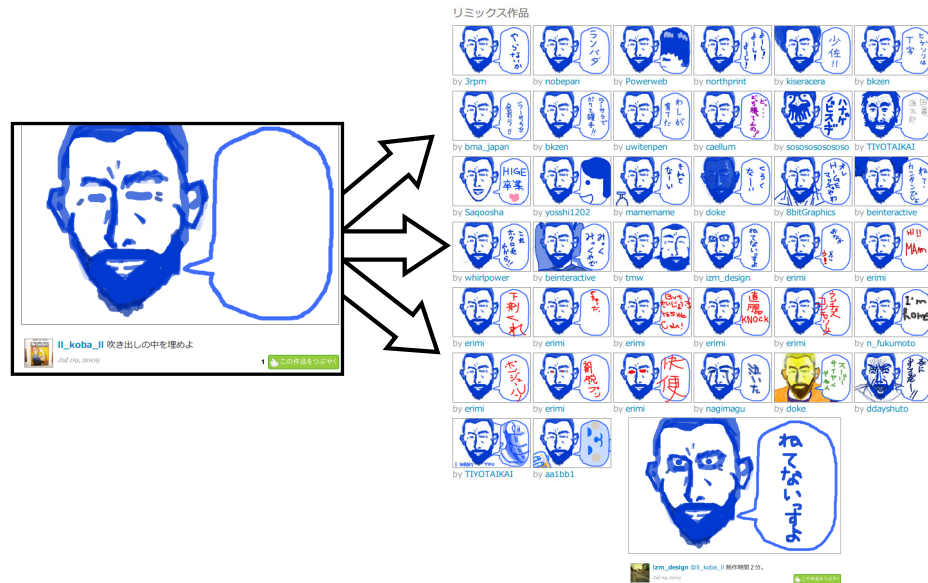


図 19 吹き出し画像からの派生 (TwitPaint)

Fig. 19 Derivations from a image containing an empty balloon (TwitPaint).

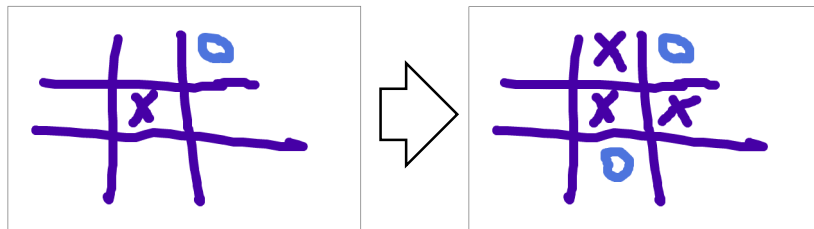


図 20 三目並べ (TwitPaint)

Fig. 20 Tic-tack-toe (TwitPaint).

派生ツリー内の子孫画像数が最も多く、最も世代数の多かった図 22 では、ある 1 人のユーザが画家 (ゴヤ) の作品を模写していた。このような力作絵画は TwitPaint の 1 回の投稿につき 140 ストロークしか描けないという制約により、1 回では描ききれない。そこで 140 ストローク描くたびに投稿して、またすぐに同じ人が派生させて描き足している。



図 21 しりとり (TwitPaint)

Fig. 21 Last and first (TwitPaint).

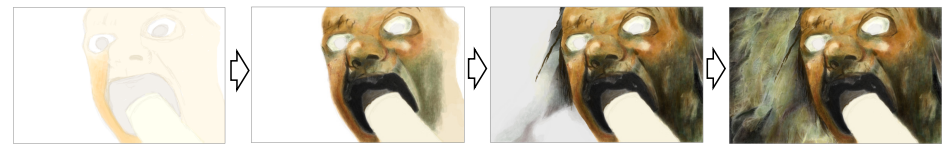


図 22 名画を 1 人で模写 (TwitPaint)

Fig. 22 Reproducing a famous painting by a user (TwitPaint).



図 23 名画を複数人で模写 (TwitPaint)

Fig. 23 Reproducing a famous painting by multiple users (TwitPaint).



一方、図 23 では有名なモナ・リザの絵を模写しているが、こちらは複数人で 1 つの絵を仕上げている。絵を仕上げようとする人もいれば、派生させた絵に落書きして遊ぶような人もおり、上記のゴヤの例に比べて枝分かれも多い。

#### 4.2.4 お 題

TwitPaint のお題機能を使った投稿では、「記憶だけでグリコのマークを描いてください」といった、記憶だけを頼りに描いて投稿させるものが特に人気であった。また「好きな歴史上の人物を描きましょう」といった、その人の好きな何かを描かせるものもお題や投稿数が多かった。上で述べた「しりとりに」や「モナ・リザを描く」のように派生を利用してパントンタッチで描くものもお題機能を使って描かれている。

投稿されたお題の数は 113 件で、お題に答える形で投稿された絵の数は 2,790 枚（全体の約 5%）であった。

## 5. 議 論

運用を通して得られたデータや事例、知見をもとに Willustrator と TwitPaint を比較し、両者の相違や特徴、問題点を議論する。さらに Web 上のイラストツールの課題や将来の方向性についても検討する。

### 5.1 機能の比較

表 2 に Willustrator と TwitPaint の機能や特徴を対比した。

#### 5.1.1 ドロー系とペイント系

最も大きな違いは描画・保存方法の違いであり、Willustrator がベクタ（ドロー系）、TwitPaint がラスター（ペイント系）となっている。

ドローツールは 1 度描いた絵を「複製する」「拡大縮小する」「色や太さを変更する」ことを得意とするため、ペイントツールに比べて再編集しやすい。他の人が絵を再利用する場合もまったく同様に再編集できるため、派生によって多くの人が絵に手を加えたあとでも、

絵の一部の色や大きさを変更するといったことができる。一方、ペイントツールでは他の人が派生させて編集する場合、元の絵に上書きする形になるため、多くの人が絵を変更し複雑な絵になると、徐々に変更が難しくなる。

また、ドローツールは直線や幾何学的な線や図形を描くのに適しているため、図のような絵を描きやすい。そして、そのような図の元となるパーツを共有し、一部だけ再利用するといったこともできる。実際に「運用と結果」で述べたように、Willustrator では作図やキャラクター素材と合成を使った絵が描かれており、TwitPaint にはない絵を生んでいる。

#### 5.1.2 編集にかかる時間

ペンの太さや色を選ぶだけですぐに絵を描けるシンプルなペイントツールに比べて、ドローツールは入力画面や操作方法が複雑になり、細かな値の調整をすることが多いため編集時間が比較的長くなる。TwitPaint は操作のシンプルさに加えて、1 投稿 140 ストローク制限により、1 つの絵を描くのにかかる時間が短い（平均 5 分 43 秒）。Twitter のようなリアルタイムに近いコミュニケーションでは、多少ラフであっても短時間で絵を描いて投稿できることが求められるため、短時間で描けるというペイントツールの特徴が Twitter を通じたコミュニケーションに適しており、TwitPaint の参加ユーザを増やした一因になったと考えられる。

### 5.2 絵の公開先とコミュニケーション

TwitPaint の他サイトからのアクセスのうち 69% が Twitter.com 経由<sup>\*1</sup>で、9% が検索エンジン経由、残りがその他のサイト経由であったことから、Twitter を用いたコミュニケーションが TwitPaint の大きな特徴であり、多くの参加者を集めた要因になったと考えられる。一方、Willustrator の開発や運営を中心に行っていた時期（2006～2007 年頃）は、まだ Twitter はユーザ数も少なく現在ほど流行っていなかった。当時は Web 上でのコミュニケーションはブログが中心であり、Willustrator も描いた絵がブログに載せられることを想定していた。この Twitter とブログという絵の公開先の違いや、Twitter に自動投稿されるというコミュニケーションへの直結度、TwitPaint のランキング（みんなの作品）、お題といった機能が、絵を通じたコミュニケーションの活性化や絵を描くモチベーション、そして参加者数の違いにつながったと考えられる。

#### 5.2.1 バイラル効果

Twitter とブログでは、人づてに情報が伝わって広まるバイラル（口コミ）効果に差があ

表 2 描画機能や特徴の対比  
Table 2 Comparing features and functions for editing.

	Willustrator	TwitPaint
描画・保存方式	ベクタ（ドロー系）	ラスター（ペイント系）
絵の特徴	図のような幾何学的で整った絵	絵画のような手描き感のある絵
描画機能の数	複雑・多機能	シンプル・低機能
編集にかかる時間	比較的長時間	短時間（派生により長時間）
派生画像の編集方法	調整/複製/合成	描き足す

\*1 Twitter の Web クライアントからのアクセスを含めると 72%以上

る。Twitterは短い発言のしやすさから、これまでWeb上に情報を発信しなかったユーザからの投稿も多い。また、更新頻度がブログに比べて多いため情報の広まるスピードが速い。

TwitPaintに絵が投稿されると、自動的にTwitterにもURLが投稿され、それを見たTwitterでつながりのある知人などに伝わる。さらにその絵が面白ければ、コメントやReTweetを通じてTwitter上で話題になり情報が広がる。また、話題になって訪問者が増えることでTwitPaintの利用者がさらに増えるという循環になっている。

一方、Twitterでは情報が広まるのは速いが、数日でタイムラインから流れてしまうため、話題が「熱しやすく冷めやすい」傾向がある。ブログでは情報の広まるスピードは遅いが、不特定多数に公開され、検索エンジンを通じて公開後しばらくしてから見られることも多いため、長期間見られるような絵に適しているといえる。

### 5.2.2 モチベーション

Twitterのバイラル効果は絵を描く人にとってのモチベーションにつながると考えられる。特に絵がうまい人にとっては、描いた絵を誰かに見てもらえることや、それが話題になることは楽しみであるため、より人に見せやすく、話題になりやすいTwitPaintはそのような人に向いている。絵がうまい人が集まることで、よりバイラル効果も高まり、派生によって作られる絵も増えるという循環を生む。

またTwitterはチャットのようにリアルタイムに近い速さでコミュニケーションできるため、投稿したあとコメントなどの反応がすぐに返ってくるという楽しさがある。この素早いレスポンスによる楽しさが、TwitPaintで絵を描く人にとってのモチベーションにつながると考えられる。

### 5.2.3 ランキングとお題を通じたコミュニケーション

TwitPaintでは、ランキング(みんなの作品)やお題も、TwitterやTwitPaint上でのコミュニケーションの活性化に貢献していた。

みんなの作品の「話題賞」「元ネタ賞」や「継続賞」では、優れた絵や面白い派生ツリーが上位に浮かび上がるようになっており、上位の絵やツリーはTwitter上でも話題になりやすい。そしてランキングで上位に載ることや、話題になることは絵を描く人にとってのモチベーションにもつながる。このように派生データはランキング手法に用いることで、話題性のある絵や力作の抽出につながり、さらにはコミュニケーションの活性化・モチベーション向上につながっているといえる。

TwitPaintのお題も、直接絵を描かない人にもリクエストを通じて参加するきっかけを与えられる点や、何を描いていいかわからない人にヒントを与える点で、コミュニケーション

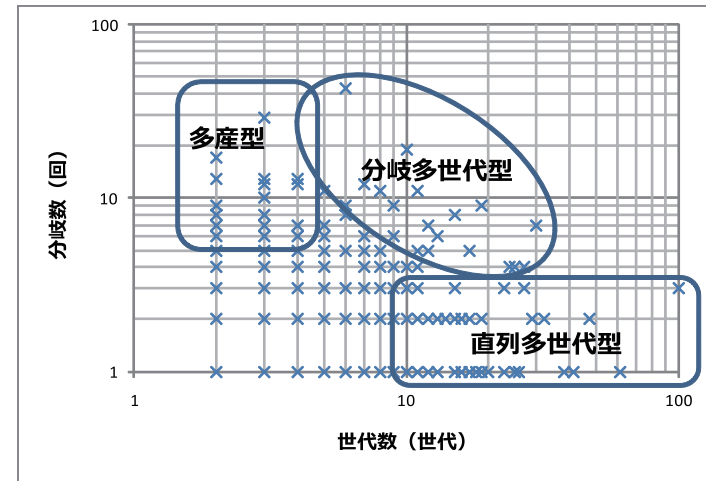


図 24 TwitPaint の派生ツリーの分類

Fig. 24 Classification of derivation trees on TwitPaint.

の活性化につながっていた。

### 5.3 派生ツリーの分類

WillustratorとTwitPaintはともに派生によって絵を描けることを特徴とし、この派生機能が活用される中で、様々な派生ツリーが形成された。派生ツリー内の分岐数<sup>\*1</sup>や世代数、ツリーの利用のされ方やユーザ層に基づいて、「直列多世代型」「分岐多世代型」「多産型」の大きく3つのタイプに分類した。図24はTwitPaintの計5,796個のツリーの分布およびそれらの分類を示すグラフである。以下にそれぞれの分類の特徴を記すとともに、派生機能を活かした表現やコミュニケーション、熟達者と非熟達者の参加について議論する。

#### 5.3.1 直列多世代型

「直列多世代型」の派生ツリーの定義は、1つの親画像に1つの子画像という関係が何世代も連続的に続くような、世代数が多く分岐が少ないツリーとする。主にTwitPaintの利用事例で述べた「力作絵画」や、漫画のようなストーリーを続けていくような利用目的にこのタイプが多かった。直列多世代型のツリーでは、絵のうまい人が少人数(1~2人)で1つ

\*1 ツリー内の各画像の子画像数の合計

の絵や話を作っていく点が特徴的である。

5,796個のツリーのうち10世代以上のツリーは55個であり、直列多世代型ツリーの数はそれほど多くない。これは、作成に関わる人が少数の絵がうまい人に限定される点と、1つの絵や話を仕上げるのに時間や労力がかかるためである。

直列多世代型ツリーは基本的に最初から最後まで少数の絵のうまい人のみが参加しており、他の人、特にあまり絵のうまい人が途中でツリーに割り込んだり、ツリーを分岐したりするといったことはあまりされない。その理由として、絵のうまい人が力作を完成させようとしている途中で、その絵の上に落書きなどをして邪魔をしまわれないようにしているためと考えられる。また、絵にあまり自信のない人が、絵のうまい人達と並んで描くことに心理的抵抗を感じることも原因として考えられる。

### 5.3.2 分岐多世代型

「分岐多世代型」は、世代数が多く、途中でときどき枝分かれもするようなツリーとする。これは主に TwitPaint の三目並べやしりとりのような遊びをする例に多く見られた。直列多世代型と異なり、分岐多世代型のツリーでは絵のあまりうまい人も参加している点や、絵そのものよりも遊びやコミュニケーションを楽しんでいるという点が特徴的である。

このような参加のしやすさに違いが生まれる理由としては、遊びで描くような絵の場合、誰でも描けるような単純な絵しか描かないといったことや、お互い絵がうまい人同士であれば、絵のうまい下手をあまり気にせず描けるため心理的抵抗が低いといったことが考えられる。

ツリーが絵のうまい人から始まると途中からの参加や分岐が難しくなるため直列多世代型になりやすい傾向にあるが、ツリー生成初期の時点で絵のあまりうまい人が参加したり、ラフな絵から始まると分岐多世代型になりやすい。つまりツリーが直列になるか分岐するかはツリー初期の時点で決まる傾向があると考えられる。

### 5.3.3 多産型

1つの親画像から多数の子画像が作られたツリーを「多産型」とする。Willustrator では図16のようにキャラクタから派生させて編集するものに多産型のツリーが見られた。TwitPaintでも同様に「吹き出しテンプレート」から派生させて絵を描くものが多かった。

Willustratorは、表2の最大世代数(5世代)が少ないことから分かるように、多世代型のツリーが少なく、多産型のツリーが多かった。これは、WillustratorはTwitPaintのような140ストローク制限がなく、また三目並べやしりとりのような遊びをするユーザが少なかったためであると考えられる。

多産型のツリーではツリーの初期にテンプレートや素材、塗り絵用の線画といった再利用されやすい絵が作られる。そのため何か少し絵に足すだけで簡単に新しい絵を作れる場合が多いため、分岐多世代型と同様に非熟達者が参加しやすい形といえる。また、同じ親/祖先の絵を各ユーザが様々な形に派生させて絵を描くことで、1つの絵がどのように変化したのかという派生の流れを見て楽しむことができるというのも特徴である。

### 5.4 Web上で使えるイラストツールの課題と展望

WillustratorとTwitPaintの運用から得られた知見をもとに、今後のWeb上で使えるイラストツールの課題や将来の可能性について検討する。

#### 5.4.1 非熟達者の継続的な利用

1章で述べたように、WillustratorとTwitPaintでは非熟達者による絵の共有や編集を目標の1つとしていた。そして、この目標のためにWeb上で簡単に共有できる仕組みや、一から絵を描かなくても派生・再編集によって絵を作れる仕組み、SNSを通じたコミュニケーションの中で気軽に絵を使えるといった仕組みを構築した。一方、図12のグラフから判断すると、1-2枚書いておしまいというユーザも多く、継続的に多数の絵を描いているユーザは少数である。さらに、こうしたユーザを観察したところ、その大半が絵の熟達者であった。

サービスを運営する中で、非熟達者が継続的に利用しにくい原因・課題がまだいくつもあることが分かってきた。克服すべき主な課題として、「絵を一般に公開することへの心理的抵抗の解消」「絵を描く人のモチベーションを高める工夫の必要性」「絵の素材とさらなる活用場面の提供」などがあげられる。以下では、このような課題をふまえて、非熟達者を含む広く一般の人のためのツールにしてゆくための施策について検討する。

#### 5.4.2 TwitterやSNSとの連携によるコミュニケーション

TwitPaintではTwitterと連携することで、優れた絵の存在が素早く人づてに広まり、また絵を描く人のモチベーション向上につながった。Twitterを含むソーシャルネットワークシステム(SNS)はWeb上での主要なコミュニケーションツールになっているため、これらの外部Webアプリケーションとの連携が今後のイラストツールにとって重要な課題となる。TwitterやSNSなどどううまく連携することで、絵の作成がより直接的にコミュニケーションにつながられるようになり、またWeb上でのコミュニケーションを絵の編集に反映できるようになると考えられる。

WillustratorとTwitPaintはともに描いた絵は基本的にすべてWeb上に公開となっていた。しかし絵に自信がない人にとっては自分の描いた絵を公開することは心理的抵抗が大きく、公開範囲が広いほどその抵抗は大きい。またTwitterでは公開範囲を細かく調節する

ことは難しいが、他の SNS と連携することで手軽かつ柔軟に公開・共有範囲を限定するための工夫も必要である。

自分の描いた絵から派生させて誰かが絵を描いたり、描いた絵に対して誰かから反応がきたりするの嬉しいものである。気軽にコミュニケーションをとりやすい知人だけに限定して公開・共有することにより、より反応が返ってきやすくするというモチベーション向上の仕組みも、絵の公開における心理的抵抗を解消するうえで重要となる。

#### 5.4.3 派生情報の分析とランキング手法

TwitPaint の「元ネタ賞」や「継続賞」では、派生情報をもとに特徴的な絵をランキング表示しており、面白い絵の発見や、絵を描く人のモチベーション向上につながった。ここでは派生機能が重要な役割を果たしていたといえる。

「絵の派生」は Willustrator や TwitPaint の持つ個性的な特徴であり、今回のように多くの絵の派生情報を記録した事例はまだ珍しく、分析の余地は大きい。今後のイラストツールにおいても、派生機能を採用し、派生情報をより詳細に記録・分析することで、新しいランキング手法などへの応用が期待できる。

#### 5.4.4 図解用素材の提供と共有

Willustrator では利用事例であげたように図解やモックアップ作成ツールとして利用されていた。図解のような用途では、より多くの人に利用機会があり、非熟達者でもパーツを組み合わせるだけで絵を描けるため、有力な発展の方向性といえる。

しかし一方で、Willustrator では図解に適した素材が少なく、必ずしも非熟達者にとって図解を作成しやすい環境ではなかった。図解用の基本的で高品質な素材をあらかじめ多数用意・提供し、さらにユーザによって描かれた素材を共有し充実させることで、非熟達者を含むより広いユーザにとって活用しやすいイラストツールになると考えられる。

また、これまでの議論で述べたような Twitter や SNS、ランキング、お題などを通じたコミュニケーションと、それらによるバイラル効果やモチベーションの向上によって、より効果的に素材が集まり、利用されることが期待できる。

## 6. 関連研究

### 6.1 Web 上で使えるドローツール

Web 上で使えるドローツールは Willustrator 以降、様々な Web アプリケーションが開発されており、特に高機能なものとして Aviary 社の Raven<sup>4)</sup> がある。ドローツールの中で作図に特化した Web アプリケーションも多く、Cacoo<sup>5)</sup> や Gliffy<sup>9)</sup>、Creately<sup>6)</sup> などが

ある。このように Web 上で利用できるドローツール、作図ツールはすでに数多くあるが、派生を活かしたドローツールは現在のところ Willustrator のみである。

### 6.2 Web 上で使えるペイントツール

Web 上の掲示板やチャット上で使えるペイントツールは比較的古くからあり、しいペインター<sup>15)</sup> のような Java Applet で実装されたものが使われていた。最近では Web ブラウザで動作する高機能なペイント・画像編集ツールも数多く登場しており、Picnic<sup>14)</sup> や Adobe 社の Photoshop Express<sup>8)</sup>、Aviary 社の Phoenix<sup>3)</sup>、SumoPaint<sup>16)</sup> などがある。また、Twitter に投稿できるペイントツールには、TwitDraw<sup>17)</sup> や drawTwit<sup>7)</sup> などがあるが、他の人の描いた絵から派生させられるものは TwitPaint のみである。

### 6.3 CSCW による絵の作成

Computer Supported Cooperative Work (CSCW) の分野で、複数人で絵を描くための研究が古くから行われており、ネットワーク越しに複数人で同時に絵を描くソフトウェアとしては、Greenberg らによるペイントツールの GroupSketch<sup>10)</sup> やドローツールの GroupDraw<sup>11)</sup> がある。

また Web 上で利用できる非同期のシステムとしては Web コンテンツへのアノテーションシステムの Annotea<sup>12)</sup> がある。Annotea そのものは RDF を用いて Web 上に様々なアノテーションを付けるためのシステムであるが、Amaya<sup>\*1</sup> というブラウザを使うことで Web 上に SVG を書き込んで協同で編集することができる。

Web 上で同期、非同期を両立したシステムに永田の Nota<sup>13)</sup>、\*2 がある。Nota ではドラッグ&ドロップによる図形の配置や、フリーハンドで線を引けるといった直接編集操作により簡単に Web ページを作成でき、さらに複数のユーザで同時編集することで Web 上での情報共有やコラボレーションを支援することができる。

Willustrator や TwitPaint のような派生型のシステムも、複数人がコンテンツ作成に関わるという点では一種の CSCW であるが、上記のような複数人で共有された 1 つのコンテンツを作成するシステムとは利用の仕方が異なる。派生型のシステムでは 1 つの絵は基本的に 1 人で編集するが、派生させることでより手軽に絵を描けるようになり、また派生させることによってユーザ間や絵の間で緩いつながりが生まれる。

\*1 <http://www.w3.org/Amaya/>

\*2 <http://nota.jp/>

#### 6.4 コンテンツの派生

アーティストの安齋らは連画<sup>1)</sup>,<sup>\*1</sup>プロジェクトで、他の人の描いた絵から連想することで、参加者が次々と紙の上に絵を描き、線をつないでいくというワークショップを行っている。さらに、このワークショップで作られた絵をコンピュータに取り込むことで、派生の様子をたどって見られるビューアを開発した。Willustrator や TwitPaint の絵がツリーのようにならんと派生していくイメージはこの連画に近い。

Processing<sup>\*2</sup>を拡張したプログラミング環境の Share<sup>2)</sup> では、プログラミングをするアーティストがソースコードを他のアーティストと共有でき、簡単に引用することができる。さらにすべてのコードの引用を記録し、グラフ状に可視化することで、多くの人に参考にされた有用なコードを発見することができる。

Web ブラウザ上で Flash を作成・共有することのできる Wonderfl<sup>18)</sup> では、プログラムのソースコードを共有し、さらにそれを他の人が派生 (Fork) させて新しいプログラムを書くことができる。Flash を使った作品を Web 上で編集・共有し、派生させながら作品を進展させるという利用スタイルは Willustrator や TwitPaint によく似ている。

#### 7. ま と め

本研究では、Web ブラウザ上で絵を編集でき、他の人の描いた絵から派生させることのできる 2 種類の Web アプリケーション: 「ドローツールの Willustrator」と 「Twitter と連携するペイントツールの TwitPaint」を開発した。さらに両ツールを長期運用し、得られたデータや事例、知見などをもとに相違や特徴、問題点の分析を行った。

ドローツールとペイントツールという描画方法の違いや、ブログと Twitter という絵の公開先の違いから、ユーザ数や画像の枚数、描かれる絵に大きな差が生じた。TwitPaint は Twitter のバイラル効果によって素早く情報を広め、さらに絵を描く人のモチベーションを高めたことが利用者の増加につながったと考えられる。

両ツールに共通する特徴である派生に関するデータや事例を分析することで、派生ツリーは大きく「直列多世代型」「分岐多世代型」「多産型」の 3 つに分類することができ、派生ツリーの種類ごとに利用目的やユーザ数、ユーザ層が異なることが分かった。

そしてこれらの知見をもとに、今後の Web 上で使えるイラストツールの課題や将来の可

能性について検討した。Twitter や SNS との連携によって絵の作成をより直接的にコミュニケーションにつなげることや、派生情報のより詳細な分析によって新しいランキング手法や可視化手法を実現すること、図解用素材の提供と共有によって非熟達者にとって活用しやすいイラストツールにすることなどが今後の課題であり、我々の目指す方向性である。

#### 参 考 文 献

- 1) 安齋利洋, 中村理恵子: 連画コラボレーションを支援するパノラマ空間ペイントシステム— The Wall, *IPSJ SIG Notes*, Vol.2000, No.13, pp.57–64 (2000).
- 2) Assogba, Y. and Donath, J.: Share: A Programming Environment for Loosely Bound Cooperation, *Proc. 28th International Conference on Human Factors in Computing Systems*, Atlanta, Georgia, ACM, pp.961–970 (2010).
- 3) Aviary: Phoenix. <http://www.photoshop.com/>
- 4) Aviary: Raven. <http://aviary.com/tools/vector-editor>
- 5) Cacao. <http://cacao.com/>
- 6) Creately. <http://creately.com/>
- 7) drawTwit. <http://drawtwit.com/>
- 8) Express, A.P. <http://www.photoshop.com/>
- 9) Glify. <http://www.glify.com/>
- 10) Greenberg, S., Roseman, M. and Webster, D.: GroupSketch, *ACM SIGGRAPH Video Review: Special Edition of the CSCW '92 Technical Video Program*, Vol.87 (1992).
- 11) Greenberg, S., Roseman, M., Webster, D. and Bohnet, R.: Issues and experiences designing and implementing two group drawing tools, *Proc. Hawaii International Conference on System Sciences*, Vol.4, Kuwaii, Hawaii, IEEE, pp.138–150 (1992).
- 12) Kahan, J. and Koivunen, M.-R.: Annotea: An Open RDF Infrastructure for Shared Web Annotations, *Proc. 10th International Conference on World Wide Web*, Hong Kong, ACM, pp.623–632 (2001).
- 13) 永田周一: NOTA: ユーザ主導型コミュニティ活動支援システムの研究, SFC-MT2006-004, 慶應義塾大学湘南藤沢学会 (2007).
- 14) Picnic. <http://www.picnik.com/>
- 15) しいペインター. <http://hp.vector.co.jp/authors/VA016309/>
- 16) SumoPaint. <http://www.sumopaint.com/app/>
- 17) TwitDraw. <http://twitdraw.com/>
- 18) Wonderfl. <http://wonderfl.net/>

(平成 22 年 6 月 28 日受付)

(平成 23 年 1 月 14 日採録)

\*1 <http://www.renga.com/>

\*2 <http://processing.org/>





神原 啓介（正会員）

1982年生。2004年慶應義塾大学環境情報学部卒業。2005年度下期未踏ソフトウェア創造事業スーパークリエイター認定。2006年慶應義塾大学大学院政策・メディア研究科修士課程修了。同年株式会社はてな入社。現在、お茶の水女子大学お茶大アカデミックプロダクション特任リサーチフェロー。ヒューマンコンピュータインタラクションの研究に従事。コラボレーションを促すWebアプリケーション開発に興味を持つ。



永田 周一（正会員）

1982年生。2005年同志社大学法学部法律学科卒業。2003年度IPA未踏ソフトウェア創造事業スーパークリエイター認定。2007年慶應義塾大学大学院政策・メディア研究科修士課程卒業。Creative Commons等のオープンコンテンツ制度やコラボラティブな創作環境の構築に興味を持つ。



塚田 浩二（正会員）

1977年生。2000年慶應義塾大学環境情報学部卒業。2005年同大学大学院政策・メディア研究科博士課程修了。同年独立行政法人産業技術総合研究所研究員。2008年4月より、お茶の水女子大学特任助教。2010年10月より、科学技術振興機構さきがけ研究員（兼任）。ユビキタス・インタフェースの研究・開発に従事。プロトタイピング、ガジェット収集・発明に興味を持つ。博士（政策・メディア）。