

## Poisson Image Editing を用いたキャラクタコラージュシステムの開発

渡辺賢悟<sup>†</sup> 伊藤和弥<sup>††</sup> 近藤邦雄<sup>‡</sup> 宮岡伸一郎<sup>‡</sup>

<sup>†</sup>東京工科大学片柳研究所 メディアテクノロジーセンター

<sup>††</sup>株式会社アイヴィス

<sup>‡</sup>東京工科大学メディア学部

## Character Collage System using Poisson Image Editing

Kengo Watanabe<sup>†</sup>, Kazuya Ito<sup>††</sup>, Kunio Kondo<sup>‡</sup>, Shinichiro Miyaoka<sup>‡</sup>

<sup>†</sup>Tokyo University of Technology, Katayanagi Institute Media Technology Center

<sup>††</sup>IVIS, INC

<sup>‡</sup>Tokyo University of Technology, School of Media Science

kengo@mediatelier.net, miyaoka@media.teu.ac.jp

### アブストラクト

キャラクタデザインは、ゲームやアニメといったコンテンツ制作において重要な作業である。デザイン作業は複数人のアイデアを、コミュニケーションを介してまとめながら進められるが、絵が描けるデザイナーと描けない者の中でアイデアの視覚化の能力に差があるため、アイデアを交換・共有するのが難しい現状がある。本研究では、絵が描けない者のアイデアの視覚化を支援するため、絵画手法の1つであるコラージュに着目する。複数の既存画像から一部分を切り出し、組み合わせるだけの作業で視覚化を行えるシステムを提案する。画像合成処理にPoisson Image Editingを用いることで、高品質なコラージュ結果を実現する。また、合成部品の切り取りを簡単にするため、部品領域の最適化処理を実装する。作成したシステムと、従来のソフトウェアを使用して視覚化した結果を比較し、システムの実用性を検証する。

### Abstract

Character design is very important work on video-game and animation contents production. However, in the case there are members who don't have character drawing skill, it's difficult for members to communicate and share character image each other. In this paper, we present a character collage system for solving this problem. In this system new character image is created easily by combining existing characters. Poisson Image Editing is applied for seamless image composition and we propose a technique for optimizing character-parts region used for the image composition. Experimental results clarify our system is effective and efficient comparing a existing software.

## 1. はじめに

ゲームやアニメといった、多様なメディアにおけるコンテンツ産業の充実がめざましく、多くの作品が制作されている。これらコンテンツは特徴あるキャラクタによって彩られており、コンテンツ全体の出来を支える最重要の要素の一つである。しかしこれらのキャラクタ制作には多くの人間と労力を必要としており、必ずしも効率的とはいえない環境で制作が行われている。将来のコンテンツ産業のさらなる充実を考えるうえで、キャラクタ制作環境の改善が必須の課題となっている。特に問題となるのが、キャラクタのビジュアル面のデザインにおけるリメイク(やり直し作業)である。

キャラクタデザインは1人もしくは複数人のメンバ全員でアイデアを出し合い、設定やビジュアルのデザインを確定していく。そのメンバの中に、デザインを視覚化する「デザイナー」、最終的にデザインの採用を決定する「プロデューサ」が含まれる。

個人制作や同人作品で1人が全て兼任する場合、アイデアがそのまま視覚情報に反映されるため、問題は発生しない。しかし、TV アニメやコンシューマゲームなどの規模の大きな作品の場合、キャラクタデザインに関わるメンバも複数となるため、全員のアイデアを、コミュニケーションを介してまとめ、デザインに反映する必要がある。このとき、コミュニケーションに用いられているのは、主にリテラル資料(文字や言葉)である。このとき、視覚化の手段を持つのはデザイナーのみであり、アイデアの視覚化についてはデザイナーと他のメンバの間で伝達力に格差が生じる。

特に、コンテンツの方針を決定するプロデューサにデザイン経験がない場合が多く、アイデアの視覚化手段がないため、意図を上手く伝えられず、デザインに上手く反映できない。結果、リメイクを繰り返すことになり、デザイナーに大きな負荷をかける原因となっている。

以上の状況から、キャラクタデザインに関わるメンバ全員が自らのアイデアを直接視覚化し、コミュニケーションに利用できることが望ましい。しかしデザイン経験のない参加者が、視覚化のために新たに描画能力を習得するのは負荷が大きい。金子ら<sup>[1][2]</sup>も、デザインの意図伝達を支援する「キャラクタデザインのための思考にあった操作や機能」を持つシステムが存在しないことを挙げており、簡易にアイデアを視覚化できるシステムの登場が期待されている。

茂木らはデザイン経験にかかわらず、アイデアの発想者自身がキャラクタの絵を作成できるようなキャラクタデザイン制作システムの仕組みを提案<sup>[3][4]</sup>しており、デザイン経験のない者がデザインを視覚化する手法について言及している。一方で、伊藤らは Poisson Image Editing 手法を用いたキャラクタのパーツ合成方法の提案<sup>[5]</sup>を行っており、良好な合成結果を得ている。

これらの流れを受け、本研究ではキャラクタについてのアイデアの視覚化を容易にするため、画像のコラージュ手法に着目する。コラージュは、既存のイメージの切り貼りによって作品を構成する絵画手法で、デザイン経験の少ない者でも簡単に導入できる手法として有効であると考えられる。特に、プロデューサと

デザイナーが別の人間となるコンテンツの場合、決定者であるプロデューサのアイデアが視覚化されることで伝わりやすくなるため、迅速にデザインを確定することが期待できる。

本研究では、既存画像の組み合わせで、簡単に視覚化作業ができる画像コラージュシステムを実装し、デザイン経験に関係なくアイデアを視覚化する手法を提案する。このシステムを利用することで、キャラクタデザインに関わる全ての構成員が、アイデアを視覚化できるようにし、デザイナーと他メンバの間にある伝達力の格差の解消を目指す。

システム内でキャラクタパーツをコラージュする作業に、伊藤らが用いた Poisson Image Editing を利用し、短時間で質の高いコラージュ結果を得られる方法を提案する。

最後に実装したシステムでコラージュを行った結果と、既存のソフトウェアでコラージュを行った結果を比較することで、アイデアの視覚に有効であるかを評価し、システムの実用性を検証する。

## 2. 関連研究

コンテンツ制作の発想を支援する研究<sup>[6]</sup>があるが、これらは文字や既存の画像を整理することで、デザインに参加しているメンバのアイデア想起を促すことに主を置いている。同様に KJ 法やブレインストーミングをベースとした発想支援のソフトウェア<sup>[7][8]</sup>は数多く存在する。これらは既存画像を編集・合成する点は考慮されておらず、発想したアイデアを意図にしたがって詳細に視覚化し、出力する手段について言及していない。

アイデアの出力について、茂木らの研究<sup>[4]</sup>では、キャラクタデザイン支援システムを実装し、デザイン経験に関係なくキャラクタを作成できるようにするための提案を行っている。この研究も前述の発想支援ソフトと同様、大量の既存画像からのイメージ選定に関しては一定の成果を得ている。アイデアを可視化する作業にコラージュを用いる点を言及しているが、デザイン経験のないユーザが考慮されておらず、成果が出ていない。

一方、画像合成システムという視点では、モーフィングを用いて複数画像間で顔画像を合成するシステム「FUTON」<sup>[9]</sup>、3DCG と実写の合成システム「AmazeART」<sup>[10]</sup>などが開発されている。このように、すでに実用化された画像合成システムは存在するが、実写を対象としたシステムがほとんどで、キャラクタ画像の合成に適した画像合成システムは存在しない。

キャラクタデザインに注目した合成処理という点では、伊藤らの研究<sup>[5]</sup>がある。Poisson Image Editing によるパーツ合成について提案しており、高品質な合成結果が得られることを確認している。しかし手法の効果についての言及にとどまっており、実用的な画像合成システムの実装には至っていない。

また Poisson Image Editing による画像合成表現を応用した研究として Diffusion Curves<sup>[11]</sup>と Gradient-Domain Painting<sup>[12]</sup>があげられる。Diffusion Curves は自由曲線の操作によって多彩なグラデーションを生み出しており、Gradient-Domain Painting では、勾配空間の編集による新しい描画機能を実現している。いずれの研究も、Poisson 方程式をベースにした新しい表現ツールとし

て注目されている。

その他、Adobe社のPhotoshop CS5のデモ<sup>[13]</sup>において、Seam Carving<sup>[14]</sup>を用いたシームレス合成処理が実装されることが発表されており、Poisson Image Editingの手法と合わせて、画像合成手法の高度化・多様化が進んでいることがわかる。

このような流れの中、本研究は、キャラクターのビジュアルデザインという特徴的な画像を対象とする点、「絵が描けない者」向けにカラージュ作業にスポットを当てた専用システムを実装する点、そしてPoisson Image Editingの新たな活用先の開拓、以上3点の特徴を持つ研究として位置付けることができる。

### 3. コラーージュシステムの設計

#### 3.1 既存のアイデア視覚化の手法

デザイン経験のない者の多くは、アイデア伝達に文字や言葉を用いる。既存の画像でサンプル提示することもあるが、実際のアイデアとの相違が誤解を招きやすい。そして、そもそもカラーージュを行って絵を示す、という考え自体が浸透していない。

仮にソフトウェアを用いてカラーージュ作業を行おうとしても、簡易なカラーージュ作業を実現したソフトウェアは存在せず、Photoshop, Gimp, SAIなどの複雑で多機能な汎用のフォトレタッチソフトを代用することになる。これらのソフトの利用は、描画経験の少ない者にとって過負荷であるが、他の選択肢がないことから、上記ソフトの利用を余儀なくされる。

よって本研究では、デザイナー経験のない者のアイデアを簡易に視覚化できるようにする点を重視し、システムを開発する。また現状でカラーージュを行う場合に用いられるであろう、上記ソフトウェアのうち、特に利用者数の多いPhotoshopとの作業比較を行いながら、カラーージュの手順を考察する。

#### 3.2 コラーージュの手順

カラーージュでは、まず下地となる画像を用意する。次にパーツ画像を読み込み、必要な加工を施した後、下地画像に貼りつける(パーツ合成)作業を繰り返し、アイデアを視覚化する(図1)。

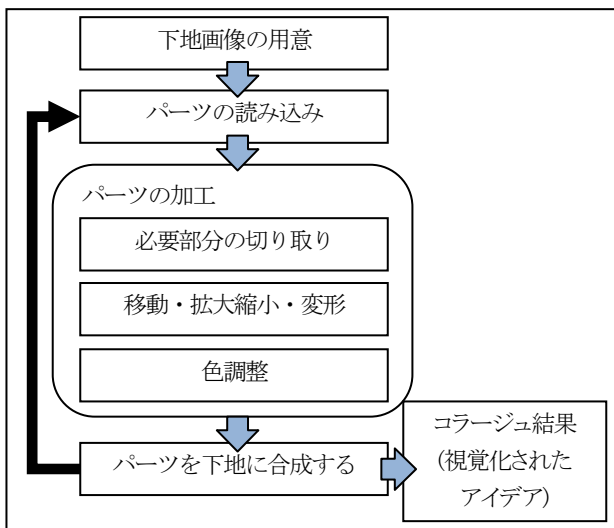


図1: コラーージュシステムの作業手順

#### 3.3 パーツの配置・パーツの埋め込み

図2のパーツを下地に合成する作業は、「パーツの配置」と「パーツの埋め込み」の2つに分類できる(図2)。

画像加工パーツの配置は、下地画像に合成した別パーツと重ならないように並べていく合成を指す。これはパーツ間の干渉がないため、単純な切り取り・変形や色調整などの画像処理で実現できる。これらの機能は既存のレタッチソフトではほぼ実装されており、利用することができる(図2(a))。

一方でパーツの埋め込みは、他のパーツに重なるため、パーツ境界部分で絵の乱れが発生する。図2(b)ではキャラクターの目を埋め込んでいるが、パーツの境界色が下地の色と合わないため違和感がでている。本研究で対象とするアイデアを視覚化したデザイン案は、デザインの完成稿のような高品質の結果を必要としないが、草案段階であっても、単純な埋め込みによる絵の乱れはデザイナーへの意思伝達を乱す可能性が高い。このような場合、既存のレタッチソフトではパーツを手作業で緻密に切り取ることで絵の乱れを防ぐが、長い作業時間が必要であり、現実的な作業環境ではない。

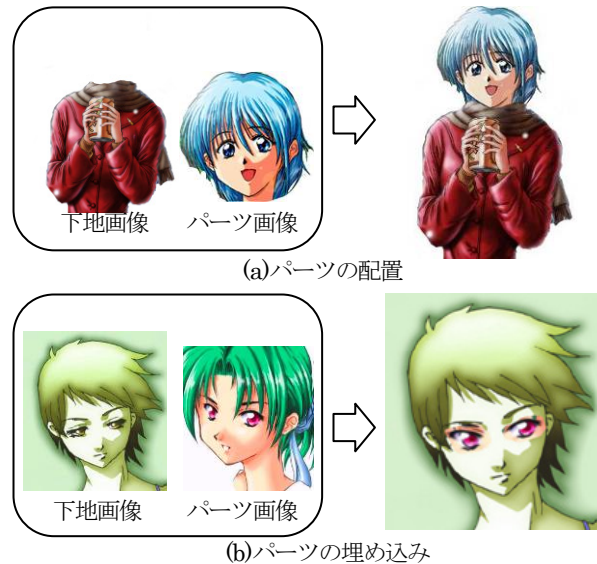


図2: パーツの配置と埋め込みの例

そこで本研究では、パーツ埋め込み時のパーツ境界部分の絵の乱れをなくすため、Poisson Image Editingによって絵をなじませる自動処理と、切り取り領域の最適化処理を実装する。これらを実装することで、ユーザーはおおまかな切り取り領域を指定する作業のみで、高品質なパーツ埋め込み結果を得ることができるようになると考えられる。Poisson Image Editingについては4章、切り取り領域の最適化処理については5章で詳述する。

#### 3.4 ユーザインタフェース

茂木らの研究<sup>[4]</sup>で試作された合成システムは、従来の画像処理ソフトウェアの利用経験がない者が利用することを前提に、意図的に従来とは異なったユーザインタフェースで実装したが、逆に作業効率の悪化につながっていたことがわかっている。これはプロデューサーにデザイン経験はなくても他のソフトウェアの利用経験があり、使用感の大きな差異から混乱を招いたため

と考えられる。

このことから、画像の移動や拡大縮小といった基本的な作業を行うユーザインタフェースには、従来のレタッチソフトの形式を採用するのが良いと考えられる。そこで日本で利用者の多い画像処理ソフト Photoshop、ペイントツール SAI を参考にしたユーザインタフェースを採用し、利用者の混乱解消をねらう。

## 4. Poisson Image Editing によるパーツ合成

### 4.1 Poisson Image Editing の原理

Poisson Image Editing<sup>[15]</sup>は、指定したオブジェクト領域  $\Omega$  内の画素において、貼りつけるパーツ画像  $f_s$  を、貼り付け先画像  $f_t$  になじむように合成する手法である。オブジェクト領域の境界部分  $\partial\Omega$  上にある貼り付け先の画素値  $f_t|_{\partial\Omega}$  を境界条件として、先に求めておいたパーツ画像  $f_s$  の内部のエッジやテクスチャの状況を表すラプラシアン  $\Delta f_s$  を反映した画像  $f|_{\Omega}$  を、Poisson 方程式  $\Delta f = \Delta f_s$  を解くことにより導き出す(図3)。

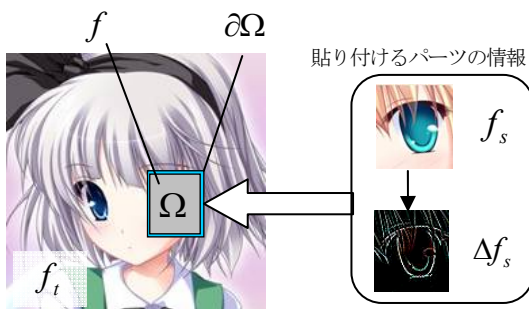


図3: Poisson Image Editing の原理

今回のシステムにおけるオブジェクト領域  $\Omega$  は、切り取ったパーツ画像の領域をそのまま利用する。つまりパーツの外縁部にある貼り付け先の画素を境界条件として Poisson 方程式を解く。

### 4.2 SOR 法

4.1 で示した Poisson 方程式  $\Delta f = \Delta f_s$  の内部領域を求める計算は、ラプラシアンを計算する離散式を用いて、式(1)のような連立方程式となる。

$$f(i, j) = \frac{1}{4}(f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1)) - \frac{1}{4}\Delta f_s(i, j) \quad \dots(1)$$

この時、式(1)について反復解法で収束解を得ることになるが、収束解を得るのに時間がかかる。そこで本研究では加速緩和係数  $\omega$  を用いて反復解法を高速化する SOR 法を用いる。式(1)の計算において反復回数  $n$  を考慮し、加速緩和係数を  $\omega$  としたとき、本研究で用いる SOR 法の式は次式(2)のようになる。

$$f^{n+1}(i, j) = (1-\omega)f^n(i, j) + \omega\left\{\frac{1}{4}(f^n(i+1, j) + f^n(i-1, j) + f^n(i, j+1) + f^n(i, j-1)) - \frac{1}{4}\Delta f_s(i, j)\right\} \quad \dots(2)$$

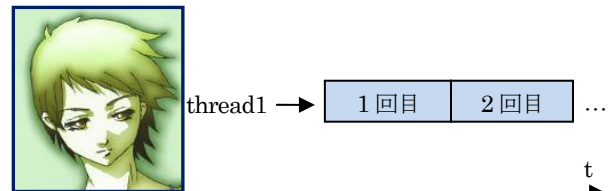
SOR 法の加速緩和係数  $\omega$  は、実験的に求める必要がある。本研究で計算が収束するまでの演算回数を検証した結果、複数の素材において収束回数が最小になりやすい  $\omega = 1.9$  に設定して用いることにした。

### 4.3 マルチコア活用による高速化

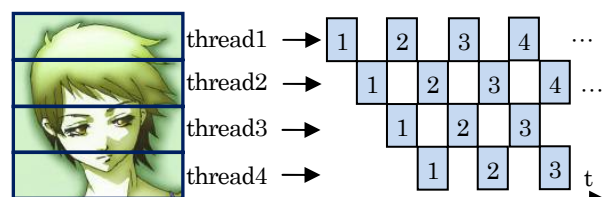
Poisson Image Editing は反復解法を用いて結果を求めるため、試作したシステム<sup>[6]</sup>において処理が遅滞し、SOR 法を用いてもなお高速化が必要であることがわかった。第一に GPGPU の並列計算による高速化が考えられるが、利用環境が限定されてしまうため、本研究のシステム目標にそぐわない。そこで本研究では今後のメニーコアの実用化を考慮し、マルチコア処理による高速化を採用する。

4.2 で示した SOR 法は、逐次計算結果を上書きしていく手法であるため、そのままでは並列計算ができない。そこで今回は反復処理の並列化を試みる。

まずコア数に合わせて処理領域を分割し、各領域に 1 スレッドを割り当てる。領域は上から順に処理していくが、同時にスレッドを処理すると、逐次更新処理の整合性が取れなくなるため、第 1 スレッドの処理が完了したことを確認したのちに、第 2 スレッドを動作させる。この流れで第 3 スレッドの処理が開始される時、第 1 スレッドは整合性を保持して実行が可能になる。よって第 1 スレッドは 2 回目の処理、第 3 スレッドは 1 回目の処理を並列で実施する(図 4(b))。



(a) シングルスレッドでの処理



(b) マルチスレッドによる並列化

図4: 並列化計算の例

## 5. パーツ領域の最適化

### 5.1 Poisson Image Editing の問題点

Poisson Image Editing はパーツ境界部分が平坦な画素値の場合、良質な結果が得られるが、パーツ境界上に画像のエッジ部分をまたいでいると、期待した合成結果が得られない。そこでユーザが指定したおおまかなパーツ領域から Poisson Image Editing に適したパーツ領域を自動的に判別する最適化処理が望まれる。

Jia らの研究<sup>[17]</sup>では、グラフの最短経路アルゴリズムで顕著なエッジを避けながら目標を切り抜く方法を提案しているが、滑らかさを欠いた切り取り結果になるなどの問題点を残している。

そこで本研究では、パーツの切り抜き処理がシステムのユーザオペレーションの一環であることに着目し、ユーザが適宜、最適化の処理を繰り返すことで、パーツ境界を少しずつ収縮していく領域収縮処理を採用する。これによって、ユーザは必要なパーツ領域を得られるまで最適化処理を繰り返しながら領域指定の微調整を行うことができ、汎用的な利用を期待できる。

## 5.2 パーツ領域境界からの領域収縮処理

初期のパーツ領域は、あらかじめユーザにおおまかに指定してもらおう(図 5(a))。この時、ユーザが選択したパーツ領域の境界部分には色値が平坦な部分だけでなく、エッジ部分をまたいだ場合などが含まれており、多彩な色値が含まれている。そのため、単色を指定した単純な領域選択処理ではパーツ領域最適化に対応できない可能性が高い。

そこで、パーツ領域の境界部分の各ピクセルを基準画素とした領域拡張処理を考える。基準画素から領域拡張を開始し、画素の RGB 値が一定の閾値範囲内(基準画素値の $\pm 10$ )の色を選択していく。この選択処理を境界部分にある全てのピクセルから行う(図 5(b))。各境界部分の基準画素の領域拡張処理によって選択された領域を全て重ね合わせ、大きな一つの領域を作成する。作成した大きな選択領域を現在のパーツ領域から除くことにより、パーツ領域の余分な選択部分を除去する(図 5(c))。ただし、選択領域には多くのノイズが含まれているため、これらのノイズを除去するために選択領域の膨張収縮処理を行う。Poisson Image Editing を適用するとき、パーツ領域が画像のエッジ部分の内部に少しでも入りこんでいると必要な部分に色が侵食してしまう。そこでエッジ部分の内部を除去しないように、パーツのエッジ部分の外縁にある領域を少し含むようにする。

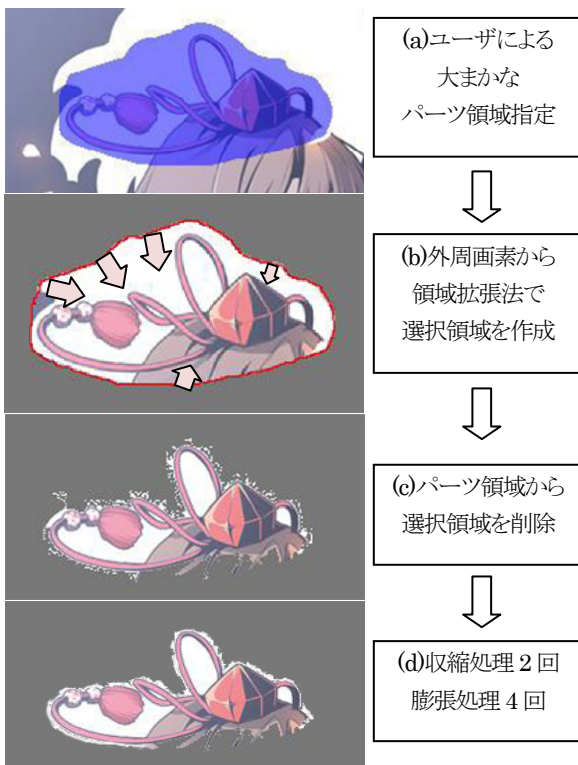


図 5: パーツ領域最適化の手順

よって、収縮処理は 2 回、膨張処理は 4 回施すようにし、パーツ画像のエッジより若干膨張した領域を維持する(図 5(d))。

ここはユーザが任意にパラメータ設定を行うという手法も考えうるが、本システムの対象である絵の描けないユーザに画像処理のパラメータ設定を行わせるのは適切ではない。またインタフェースが煩雑になると、簡易に利用可能というシステム本来の目的が失われる。そこで本システムの領域収縮処理は、閾値 $\pm 10$ 、収縮 2 回・膨張 4 回を固定のパラメータを設定する。このパラメータは、最適化処理を繰り返し行った時、輪郭の明確なキャラクターパーツにおいてパーツ領域形状がパーツ周辺で定常化しやすい特徴を持っており、用途に対して適当であるため、採用する。

## 6. システムの実装と実験・評価

### 6.1 システムの実装

本節では、提案したコラージュシステムの構成と実装したユーザインタフェースについて述べる。図 6 にシステムの全体図を示す。画面左に作業用ツール、右側にキャンバスを用意している。パーツ加工や下地加工に用いる全ての機能を画面上に表示し、ユーザがツールを操作しなくても、機能を一望できるようにした。極力操作が必要な項目を少なくすることで、新しいソフトウェアの習得時間の軽減について考慮した。また加工中のパーツには Photoshop などと同じ操作感で利用可能な変形ボックスを採用し、ユーザのソフトウェア経験の有無に関係なく、パーツの変形等を直感的にできるよう工夫した。



図 6: コラージュシステムの全体図

### 6.2 Poisson Image Editing と領域最適化の実験

パーツの埋め込み作業において、単純な埋め込み・Poisson Image Editing・領域最適化処理を利用した場合での埋め込みを行う。処理結果を比較し、提案手法の有効性を確かめる。以下の図 7 にその結果を示す。

図 7 の結果は、ユーザは大まかな選択のみ行うことを前提としている。図 7(b)は大まかに目を選択し埋め込んだ結果だが、肌の色の違いや不要な領域の埋め込みにより絵の乱れが著しい。

また同じパーツ領域で Poisson 合成を施したものが図 7(c)である。肌の色は周辺になじみ違和感は減っているが、パーツ領域が下地の眉毛などに侵入しており、合成結果に不都合が生じている。図 7(d)は図 7(b)に領域最適化を 1 度施した結果である。不要なパーツ領域が削がれ、必要なパーツ周辺がうまく切り取られている。図 7(d)のパーツ領域で Poisson 合成を行ったものが図 7(e)である。不要な領域を含んでおらず、かつ肌の色などは周辺になじみ、違和感を解消していることがわかる。以上の結果からパーツ埋め込みにおける、パーツ領域最適化と Poisson Image Editing の組み合わせ処理が有効であることがわかった。

ただし最適化の処理において、パーツの特徴やユーザが初期に選択する領域の形状によって上手くいかない例があった(図 8)。特に、目などのパーツは髪の毛や眉などの別パーツを選択領域に含む場合があり、髪の毛や眉の輪郭線が最適化を妨げることがあった(図 8(c))。また、パーツとパーツでない部分の色値が近く、明確な輪郭が存在しない場合、必要な領域まで収縮を繰り返し、領域の欠損が発生した。これらの現象を防ぐためには最適な閾値設定が必要となるが、本研究ではユーザによるパラメータの任意設定を前提としないため、閾値の設定方法に課題があることがわかった。

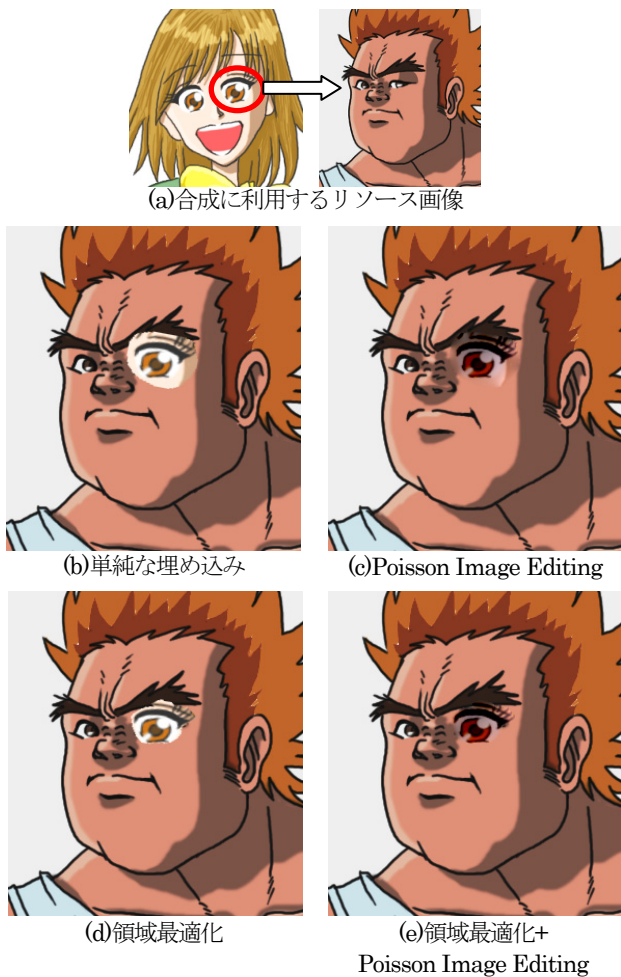


図 7: パーツ埋め込み結果の比較



図 8: 最適化処理がうまくいかない例

### 6.3 マルチコア化による高速化の実験

4.3 で示したマルチコア処理による高速化の効果を検証する。256x256, 512x512 ピクセルの画像にそれぞれ Poisson Image Editing の合成処理を 5 回施した。5 回の平均処理時間を表 1 に示す。なお、使用したマシンの CPU は Core2Quad Q9550(4 コア)である。

表 1 の通り、マルチコア対応にすることで処理速度がおよそ 1.8~2.1 倍に高速化しており、4.3 の並列化計算処理が有効であることが確認できた。ただし表 1 の時間は反復計算の終了チェック処理を含んでいるため、チェックサイズの大きい 512x512 では若干比率が低下した。今回は 4 コア CPU による結果だが、コア数が増加すると共にさらなる高速化が期待できる。

表 1: Poisson Image Editing の処理時間(5 回平均)

	256 x 256	512 x 512
シングルスレッド	690 (ms)	4191 (ms)
マルチスレッド(4 コア)	321 (ms)	2352 (ms)
速度比率	214%	178%

### 6.4 実用性・作業効率の評価

実装したカラージュシシステムを用いた場合と、既存のソフトウェアで作業を行った場合の作業効率について比較、評価する。

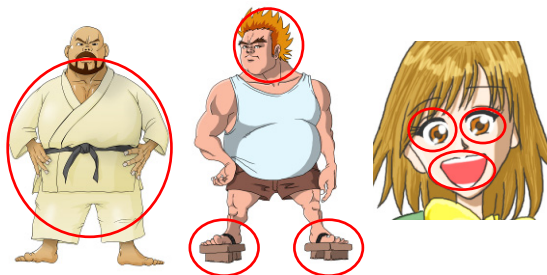
ここではキャラクターパーツの配置と、顔内部パーツの埋め込み作業を処理する。利用するソース画像と利用箇所を図 9(a)に示す。そして Photoshop CS3 で作業を行った結果が図 9(b)、カラージュシシステムで作業を行った結果が図 9(c)である。また、それぞれの作業に必要な作業時間・習得時間を表 2 に示す。なお、習得時間については Photoshop の利用経験の少ないユー

ザ5名の平均的な習得時間を計測した。また図9の作業・時間計測については Photoshop の使用経験のある筆者が行い、数度の結果の平均値を算出した。

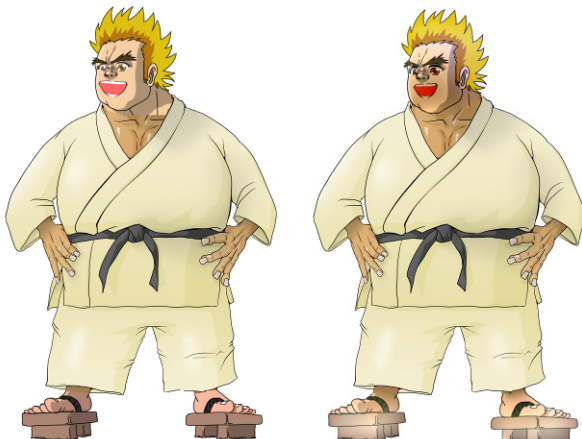
表2の結果から、本システムの利用と Photoshop の利用に必要な習得時間を比べたところ、Photoshop は自主的に作業ができるまでには数日かかるのに対し、本システムは10~15分程度の習得時間でキャラクタ作成に利用できるようになるため、デザイン経験のないユーザが導入しやすく、習得の手間が省ける簡易なシステムであることが確認できた。

さらに実際に合成作業を行う際にも、必要な時間が約半分にすることが確認できた。Photoshop では、パーツを綺麗に切り抜くための作業に多くの時間を要した。一方で本システムでの主な作業は、Poisson Image Editing で良好な結果が得られるためのパーツ位置の調整作業となった。また、他のパーツ加工作業においても、本システムを利用することで作業時間が短くなる傾向が見られた。

図9(b)と図9(c)の結果を比較すると、Photoshop の結果では、道着の男性の肌の色と、顔と足に適用した肌の色の調整が取れず、首元と足の色に違和感が出ている。図9(c)ではPoisson Image Editing によって肌の色がシームレスに合成されており、



(a)合成に利用するリソース画像



(b)Photoshop による結果

(c)本システムによる結果

図9: 合成作業のソース画像と結果画像

表2: 図9の合成作業に必要なとした作業時間

	Photoshop CS3	本システム
作業総時間	約9分	約4分15秒
習得時間	数日	10~15分
時間を要する主な作業	パーツ抜き加工 結合部の色調整	パーツ変形・ パーツ位置調整

パーツ境界部の違和感が解消されているのがわかる。Photoshop でこれらの違和感を無くすためには、色調整や、より緻密なパーツ切り抜き作業が必要となり、Photoshop での作業時間はさらに増えることになると考えられる。

上記の結果から、本システムがアイデアの視覚化において、習得時間・作業時間・結果の品質向上のいずれにおいても効果的であることが示された。一方で、本システムでは、Photoshop などで実装されているピクセル単位でアルファチャンネルを設定できるような「パーツ領域形状の微調整」を行う機能がなく、表1の作業中においても微調整作業に遅滞が発生した。今後、これらの機能を充実することによって、さらなる作業の効率化が期待できると考えられる。

また、図9(c)の結果において Poisson Image Editing によるグラデーションが発生している。これは図9(c)の「頭全体」といった大きく複雑なパーツを合成した場合に顕著にあらわれる。このグラデーションはユーザの意図を阻害する可能性がある。しかしながら本研究の目的に沿って、絵が描けないユーザが可視化手段を得るという点を考慮すると、グラデーションの悪影響に比して、導入のメリットは十分に大きいと考えられる。

さらに図7(e)のような合成の例では、良好な合成結果が得られていることから、合成するパーツの特徴によってグラデーションの影響も大きく異なる。これらの結果から Poisson Image Editing のグラデーションの特性を分析し、合成するパーツの特徴によって処理方法に工夫を加えていく必要があるとわかった。

## 7. おわりに

本研究では、キャラクタデザイン作業時における、デザイン経験のない者の発想を視覚化するためのカラージュシステムの提案・実装を行った。

既存のソフトウェアにおいて煩雑な作業になっていた「パーツの埋め込み」において Poisson Image Editing と領域最適化の手法を用いることで、高品質なデザインを効率的に作成可能であることを示した。金子らが提示した「キャラクタデザインのための思考にあった操作や機能」を備えたシステムとして、一定の成果を得ることができた。

また当初の目的であった、デザイン経験のない者がアイデアを視覚化して示せるようになることで、絵によるコミュニケーション手段を手に入れることができ、アイデアを容易に視覚的に反映できる効果的なシステムになったと考えられる。

今後はカラージュシステムの実用性を追求するため、実験過程でユーザインタフェースに不便を感じた部分の改良、必要な機能の追加等を行っていく。加えて、より高精度なパーツ領域の最適化手法や、Poisson Image Editing の応用を考え、作業効率と合成品質の向上を図りたい。

なお提案・実装を行ったシステム「Chara Collage」はweb上で公開しており、自由に利用することが可能である<sup>[18]</sup>。

## 謝辞

論文を執筆するにあたり、実験用の画像を快く提供して下さいました東京工科大学片柳研究所クリエイティブラボの岡本直樹氏に心より感謝いたします。

## 参考文献

- [1]金子満, 近藤邦雄, 岡本直樹, 三上浩司:”創作テンプレートを用いたデジタルキャラクターメイキング手法の提案”, NICOGRAPH 論文コンテスト,(2009)
- [2]金子満, 近藤邦雄, 岡本直樹, 三上浩司:”映像コンテンツ制作のためのデジタルキャラクターメイキング教育”,NICOGRAPH 論文コンテスト,(2009)
- [3] 茂木, 松本, 近藤, 金子:”リテラル資料に基づくキャラクターデザイン構成手法の研究”,第23回NICOGRAPH 論文コンテスト論文集 (2007)
- [4] 茂木, 岡本ら:”デジタルスクラップブックを用いたキャラクターデザイン原案制作システム”, 2009年度日本図学会春季大会論文集 (2008)
- [5] 伊藤, 渡辺, 宮岡: “キャラクターデザインのための画像合成手法の研究” 第71回情報処理学会全国大会講演論文集, 6R-2 (2009)
- [6]後藤田広也, 伊藤雄一, 門脇英男, 北村喜文, 岸野文郎:”映像コンテンツ作成のために発想支援システム”, エンタテインメントコンピューティング2007講演 論文集, pp. 99-102, (2007)
- [7] Mind Mapper : <http://www.mindmapper-japan.com/>
- [8] Mind Manager : <http://www.pluslink.jp/product/index.html>
- [9] 蒲池, 向田, 吉川, 加藤, 尾田, 赤松: “顔・表情認知に関する心理実験のための顔画像合成システム : FUTON System “, 電子情報通信学会技術研究報告. HIP, 97(509) pp.73-80 (1998)
- [10] Amaze ART: <http://www.ntt-at.co.jp/product/amazeart/>
- [11] Orzan, A. et al. Diffusion Curves: “A Vector Representation for Smooth-Shaded Images”, Proc. SIGGRAPH2008, pp.92:1-92:8 (2008)
- [12] McCann, J, and Pollard, N. “Real-Time Gradient-domain painting”, Proc. SIGGRAPH2008, pp.93:1-93:7 (2008)
- [13] New image editing tool for PhotoshopCS5: <http://cs5.org/>
- [14] Avidan, S. and Shamir, A. “Seam Carving for Content-Aware Image Resizing”, ACM Transaction Graph, Vol.26, Number.3, SIGGRAPH2007 (2007)
- [15] P'EREZ, et al. ”Poisson image editing” SIGGRAPH 2003, pp313-318 (2003)
- [16] 渡辺, 伊藤, 茂木, 岡本, 近藤, 宮岡:”Poisson Image Editingを用いたキャラクターカラージェシシステムの開発”, 第25回NICOGRAPH 論文コンテスト論文集 (2009)
- [17] Jia, J., et al. “Drag-and-Drop Pasting” SIGGRAPH 2006, pp631-636 (2006)
- [18] けん悟庵 : <http://kengolab.net/>

## 渡辺 賢悟



東京工科大学メディア学部メディア学科卒, 同大学院メディア学研究科博士前期課程修了. 現在東京工科大学片柳研究所メディアテクノロジーセンター嘱託研究員.

## 伊藤 和弥



2009年東京工科大学メディア学部メディア学科卒業. 同年株式会社アイヴィス入社. 現在に至る. 日米共同防衛省画像ジャイロ開発に従事.

## 近藤 邦雄



東京工科大学メディア学部教授, 工学博士(東京大学). 名古屋工業大学卒業, 名古屋大学教養部, 東京工芸大学, 埼玉大学工学部情報システム工学科などで勤務. コンテンツ工学, コンピュータグラフィックス, アニメやゲームのためのデジタル映像制作システム開発などの研究に従事.

## 宮岡 伸一郎



東京工科大学メディア学部教授. 工学博士(京都大学). 京都大学工学部数理工学科卒, 同大学院工学研究科修士課程修了の後, 日立製作所に勤務. 同社システム開発研究所主任研究員, 独立しシステムコンサルタントを経て現職. イメージメディア関連の教育・研究に従事.