

# インタラクティブな勾配編集による Poisson 画像合成手法

岡良祐<sup>†</sup> 渡辺賢悟<sup>††</sup> 宮岡伸一郎<sup>††</sup>

<sup>†</sup>東京工科大学大学院 バイオ・情報メディア研究科

<sup>††</sup>東京工科大学 メディア学部 メディア学科

Poisson Image Composition based on Interactive Gradient Editing

Ryousuke OKA<sup>†</sup>, Kengo WATANABE<sup>††</sup> and Shinichiro MIYAOKA<sup>††</sup>

<sup>††</sup>Graduate School of Bionics, Computer Science and Media Science, Tokyo University of Technology

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

E-mail: [aquaragangel@msn.com](mailto:aquaragangel@msn.com) [kengo@mediatelier.net](mailto:kengo@mediatelier.net) [miyaoka@media.teu.ac.jp](mailto:miyaoka@media.teu.ac.jp)

## 1. はじめに

近年、Poisson 画像合成<sup>[1]</sup>を初めとして勾配情報をベースにした画像編集手法の研究が活発化している。ドラッグ&ドロップ<sup>[2]</sup>やブラシインタフェース<sup>[3]</sup>などで編集するものが主流だが、合成する画像の範囲・境界の設定によっては意図した合成が出来ない場合もある。これに対し、境界最適化手法<sup>[2]</sup>が提案されている。しかし、Poisson 画像合成は境界値が色調に大きな影響を与えるため、想像と異なる色調が得られることも多く、境界最適化だけではユーザの望む結果が得られない場合がある。特にイラスト画像は境界情報・勾配情報が自然画像とは異なる特徴があり、Poisson 画像合成の色調変化に与える影響は著しい。

本手法では上記の Poisson 画像合成が生み出す色調の不具合について、特に対象をイラストに限定し解決策を模索する。ブラシインタフェースを主とし、勾配情報をユーザに直観的に編集させる。また、勾配を編集した結果を随時表示することによりインタラクションを確立し、ユーザが結果を見ながら即時に微調整を行えるよう工夫し、実用性の高い勾配の編集ツールの実装を目指す。

## 2. 関連研究

従来の輝度空間における画像合成では、合成する画像と合成先の画像の色調が合わないことがある。Photoshop や Painter などといった画像処理ソフトの機能を使うことにより、色調の違いを解消することが可能である。しかし、このような画像処理ソフトは多機能なソフトであるため、使い慣れていない人には不便である。そのため、直観的な作業で違和感のない画像合

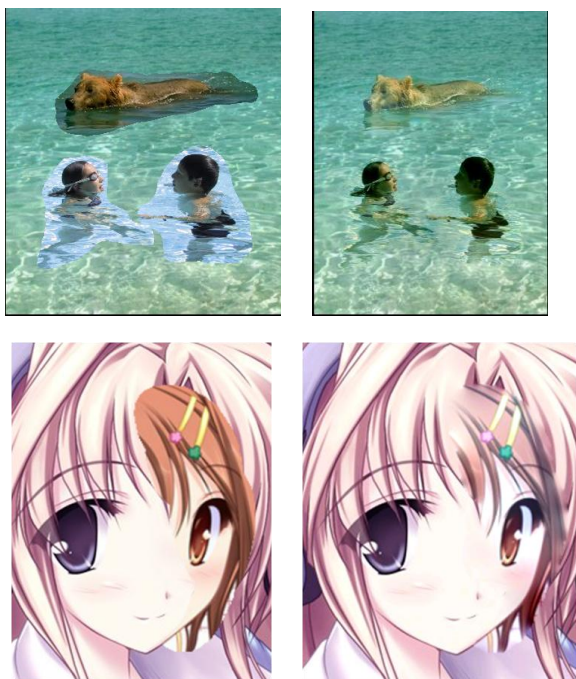
成ができる必要がある。そして、合成時に起きる微調整も直観的な作業で出来ることが望ましい。従って、次の項目を本研究の対象とする。

- 合成時の色調の違いの解消
- ドラッグ&ドロップによる合成
- 直観的な微調整が可能

従来の画像の貼り付けのような画像合成では、輝度値を上書きするため、色調に違いが表れてくる。そこで、Perez 氏らの研究<sup>[1]</sup>では、Poisson 方程式を利用した画像合成手法を提案している。この手法では、従来の輝度空間での編集ではなく、勾配空間における画像合成を行っている。これにより、画像合成時に生じる色調の違いを感じることなくシームレスな画像合成ができる。

しかし、この Poisson 画像合成は常に最適な合成を行うことができるわけではない。この原因の1つに、合成する際に画像の用いる境界部分の輝度差によるものがある。そこで、Jia 氏らの研究<sup>[2]</sup>では、この Poisson 画像合成時に生じる境界部分の輝度差が最小となるように境界の最適化を行うものである。これにより、結果画像の色調の違いを改善することができる。

一方、Poisson 画像合成をブラシインタフェースとして利用したものが、McCann 氏らの研究<sup>[3]</sup>である。この研究では、画像の勾配情報をブラシのテクスチャ情報として勾配空間内で編集する。既存の Photoshop®の修復ブラシは輝度値を上書きする処理を行うが、この研究では勾配情報を編集し Poisson 方程式を解く処理を行う。これにより、修復ブラシよりも違和感のない編集が可能となる。



通常合成

Poisson 画像合成

図 1. Poisson 画像合成

また、宮岡氏の研究<sup>[4]</sup>では、画像から得られる勾配情報を分割する。その勾配情報を編集した後、Poisson 方程式を解き基本画像を生成し、これを利用して画像を再構成する手法を提案している。この手法により、画像のイラスト風変換、グラデーション抽出、線の抽出・消去などが可能なことが示されている。

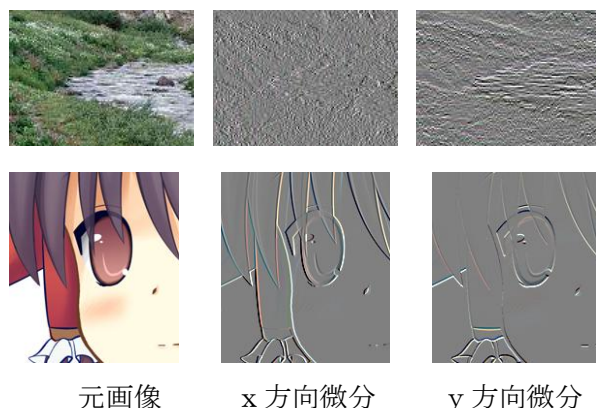
### 3. Poisson 画像合成とその問題点

ソース画像を $f_s$ 、ターゲット画像を $f_t$ 、合成領域内部 $\Omega$ 、合成境界 $\partial\Omega$ とすると、Poisson 画像合成は以下の Poisson 方程式の Dirichlet 境界値問題を解くことによって結果画像 $f$ を得ることが出来る。このとき、 $\mathbf{g}$ は $f_s$ の勾配情報である。

$$\nabla^2 f = \nabla \mathbf{g} \text{ on } \Omega, \quad \text{with } f|_{\partial\Omega} = f_t|_{\partial\Omega} \quad \dots(1)$$

Poisson 画像合成は、境界 $\partial\Omega$ でターゲット画像の輝度値に等しく、領域内部 $\Omega$ ではソース画像のラプラシアンに一致するようになっている。Poisson 画像合成の実写画像とイラスト画像での合成例を図 1 に示す。

しかし、従来の Poisson 画像合成では、合成範囲の設定によっては意図した結果が得られないことがある。例えば、ソース画像内にエッジの少ない、またはエッジ強度が弱い部分がある



元画像

x 方向微分

y 方向微分

図 2. 各画像の勾配

場合、その部分に Poisson 画像合成特有の不具合が発生する。これは Poisson 方程式を解く際に用いられる境界値と勾配情報が影響するため、境界値と勾配情報を編集し調整する必要がある。しかし、境界値を編集してしまうと境界部分に色の違いが出てシームレスな合成が維持できなくなる。そのため、本手法では勾配情報を編集ことにする。

ここで、勾配編集の 1 つであるブラシインタフェースを用いた手法<sup>[3]</sup>がある。この手法では、勾配編集を行う際に勾配方向を考慮した強調・抑制する機能が提案されている。この機能により、コントラストの強調やライティングの処理が行える。しかし、勾配を選択せずに一様に強調・抑制を行うため、余計な勾配も編集を行ってしまう。

また、本研究で対象とするイラスト画像には自然画像とは異なり、エッジやグラデーションがはっきりと存在している。図 2 の上段が自然画像、下段がイラスト画像の勾配情報で、左側から元画像、x 方向微分、y 方向微分である。これからわかるように、エッジ部分の勾配が強くなるのが分かる。イラスト画像の場合、平坦な領域と強いエッジの部分の差が顕著であり、それぞれに対して勾配編集する必要がある。

ブラシインタフェースを用いた勾配編集を行う際に一様な強調・抑制を行うのは、意図しない勾配まで編集してしまうことがある。そこで、直観的に勾配を編集できるようにするため、勾配を選択し、適切に編集する必要がある。

### 4. 勾配情報の編集

本研究では、合成する画像の勾配情報を編集することにより微調整を行えるようにする。この時に得られる編集された勾配情報を次式のように定義する。

$$\begin{aligned} G^x(i, j) &= w(i, j) \cdot g^x(i, j) \\ G^y(i, j) &= w(i, j) \cdot g^y(i, j) \end{aligned} \quad \dots (2)$$

このとき、 $G^x$ と $G^y$ はそれぞれ編集された勾配情報、 $g^x$ と $g^y$ はそれぞれソース画像 $f_s$ の勾配情報、 $(i, j)$ は編集する座標、 $w$ は勾配情報に対する重みである。そして、(1)式を書き直したものが次式(3)になる。

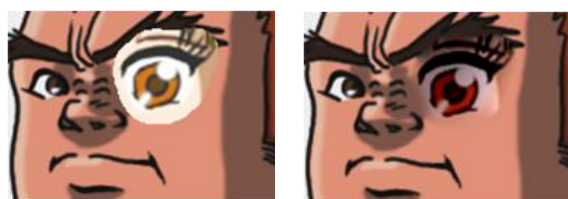
$$\nabla^2 f = u \quad \dots (3)$$

(3)式を解くことにより結果画像 $f$ を得る。このとき、 $u$ は(2)式を用いて、(4)式のように計算ができる。

$$\begin{aligned} u(i, j) &= G^x(i, j) - G^x(i-1, j) \\ &\quad + G^y(i, j) - G^y(i, j-1) \end{aligned} \quad \dots (4)$$

#### 4-1 ブラシインタフェイス

勾配編集の方法にブラシインタフェイスを提供する。これは、ユーザが望んだ部分を直観的に編集できるようにするためである。しかし、ブラシインタフェイスによる編集を行うと、ユーザの意図しない勾配も編集してしまうことがある。

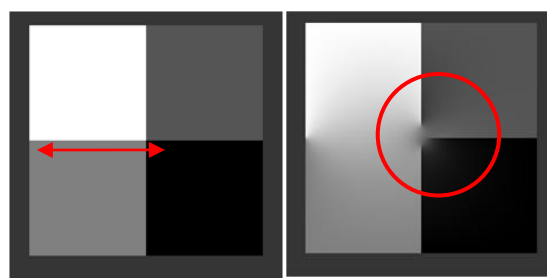


通常合成      Poisson 画像合成

図 3. Poisson 画像合成

図 3 はイラスト画像に Poisson 画像合成を適応した例である。合成後の画像から分かるように白目の部分が暗くなり、目じりの部分にソース画像の余計なエッジが出ているのが分かる。このような不具合は、渡辺氏らの研究<sup>[5]</sup>でも着目している。本研究では、この問題を合成部分の勾配編集によって解決することを試みる。

編集する勾配をブラシインタフェイスの提供によりユーザの望んだ部分を編集できるようにする。まず、図 4(a)にあるサンプル画像に赤い矢印方向に白と灰色の境にある横方向のエッジを除去するように勾配編集を行ったものである。その結果が図 4(b)で、中央の赤丸で囲まれた部分で意図しないエッジも編集されているのが分かる。そのため、編集する勾配を選択して編集する必要がある。



(a) 元画像      (b) 編集後

図 4. サンプル画像での実験

図 4(a)のサンプル画像はグレースケール画像で、輝度値は左上が 255、右上が 85、左下が 128、右下が 0 となっている。

本手法では編集する勾配を閾値処理で適切に選択し処理を施す。(2)式の重み $w$ を次のように定義する。

$$w_n = w_{n-1} + w_b \cdot w_r \cdot w_g \quad \dots (5)$$

このとき、 $w_{n-1}$ はブラシで編集前の重みで、 $w_n$ は編集後の重みである。また、 $w_b$ は編集する範囲全体にかかる重み係数である。 $w_r, w_g$ はそれぞれブラシの半径、勾配に対する重みである。(5)式により、ブラシインタフェイスを用いた反復的な作業によって重みの編集を少しずつ行うことができる。これにより、自由に勾配を編集することが可能となる。そして、 $w_b$ の値を正の数にすることにより勾配の強調を、負の数にすることにより勾配の抑制を行うことができる。

#### 4-2 ブラシの半径による重み処理

ブラシによる編集において、編集する部分の重みをなめらかに作ることにする。これは編集した部分に違和感がないようにするためのものである。このことから、 $w_r$ を次式のように定義する。

$$w_r = 1 - \frac{(i-c_x)^2 + (j-c_y)^2}{brush\_radius^2} \quad \dots (6)$$

このとき、 $(i, j)$ は編集する座標、 $(c_x, c_y)$ はブラシの中心座標である。また、ブラシの半径を超える距離にある座標による計算の場合、(6)式は負の値とるが、その時は 0 とする。

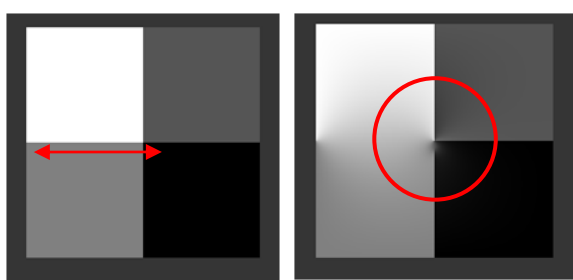
#### 4-3 勾配強度による閾値処理

ブラシによって指定された処理領域にある勾配強度と閾値 $th_m, th_M$ をもとに次式のように処理する勾配を選ぶ。

$$w_g = \begin{cases} 1 & th_m \leq \|g\| \leq th_M \\ 0 & \text{otherwise} \end{cases} \quad \dots (7)$$

このとき、 $\|g\|$ は勾配の強度である。この処理により、イラスト画像のエッジやグラデーションのみといった任意の勾配強度に対する編集が可能となる。そして、勾配強度のみの判定によって編集された結果画像は図 5(b)である。このときの閾値を $th_m = 105$ から $th_M = 153$ に設定した。

図 5(b)では、赤丸の部分にある灰色と黒の境にある縦方向のエッジまで除去されていることが分かる。よって、勾配強度のみの閾値処理だけでは不完全であるということが言える。



(a) 元画像 (b) 編集後

図 5. 勾配強度での閾値処理

#### 4-4 勾配方向による閾値処理

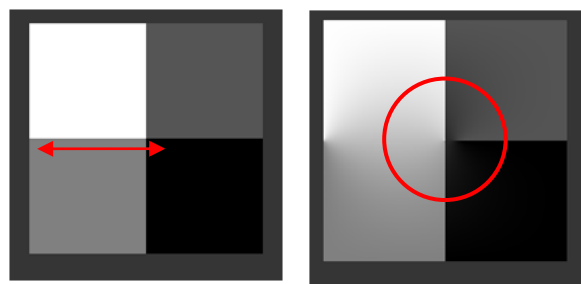
ユーザが編集する際にブラシストロークの方向ベクトル $b$ を得ることが出来る。この情報を用いて勾配方向に対する閾値処理を行う。

ブラシによって指定された編集領域内にある勾配と閾値 $th_a$ ,  $th_D$ をもとに処理する。

$$w_g = \begin{cases} 1 & th_a \leq \theta \leq th_D \\ 0 & \text{otherwise} \end{cases} \quad \dots (8)$$

このとき、 $\theta$ は $g$ と $b$ のなす角度である。勾配方向のみの判定によって編集された結果画像は図 6(b)である。このときの閾値を $th_a = 81$ から $th_D = 99$ に設定した。

図 6(b)では、図 5(b)のような輝度差による不具合は発生していないが、今回編集したい横方向のエッジとは別の同方向のエッジまで編集されているのが分かる。よって、勾配方向のみの閾値処理でも不完全であるということが言える。



(a) 元画像 (b) 編集後

図 6. 勾配方向での閾値処理

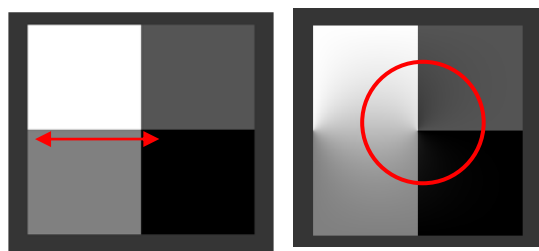
#### 4-5 勾配強度・方向による閾値処理

図 5(b)、図 6(b)から分かるように勾配強度・方向を各々で閾値処理を行う場合、閾値処理を行わない時よりも適切に処理が行われている。しかし、選択が不適切で違和感のある編集だということが分かる。そのため、勾配を強度・方向ともに判定する必要がある。

$$w_g = \begin{cases} 1 & th_a \leq \theta \leq th_D \ \& \ th_m \leq \|g\| \leq th_M \\ 0 & \text{otherwise} \end{cases} \quad \dots (9)$$

これにより、勾配強度・方向の判定によって編集された結果画像は図 7(b)である。このときの閾値をそれぞれ、 $th_a = 81$ から $th_D = 99$ と、 $th_m = 105$ から $th_M = 153$ と設定した。

図 7(b)では、図 5(b)や図 6(b)のような不具合が無く、白と灰色の境にある横方向の勾配のみ編集されていることが分かる。また、赤丸の部分で、これまで出ていた余計な部分の勾配の編集がされていないことが分かる。



(a) 元画像 (b) 編集後

図 7. 勾配強度・勾配方向での閾値処理

#### 4-6 輝度信号を用いた判定処理

図 7(b)で分かるように、勾配強度と勾配方向の両方で閾値処理を行うことにより意図した勾配にのみ編集をすることができる。

そこで、実際にイラスト画像に適用した例が図 8 である。3 章でも指摘した白目の色調の変

化と目じりにある余計な線に対して編集を行ったものである。図 8(c)は判定処理に RGB の各成分で行ったもので、不具合が生じていることが分かる。これは、閾値判定時に処理されなかったチャンネルがあったため、このような結果になった。閾値判定で RGB の各成分で行うのではなく、輝度信号 Y を用いて判定を行い、編集する勾配は RGB の各成分に適用することにする。輝度信号 Y による判定処理を行った結果が、図 8(d)である。

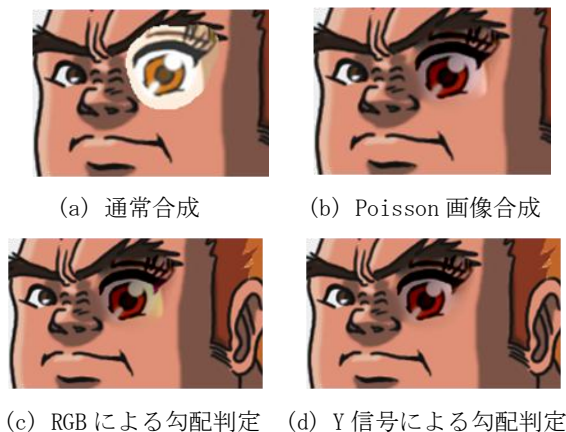


図 8. 判定チャンネルの違い

## 5. 画像の再構成

4 章では、勾配情報の編集に関して説明した。この章では、その勾配情報をもとに結果画像を生成する方法に関して説明する。

本研究では、Poisson 画像合成で得られる結果画像  $f$  は次のような手順で求める。

まず、(2)式を離散的に解くために以下の連立方程式を作る。

$$f(i, j) = \frac{1}{4}\{f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1)\} - \frac{1}{4}u(i, j) \quad \dots (10)$$

式(3)について SOR 法による反復解法で収束解を得る。このときの反復解法で用いる式は次式(4)のようになる。

$$f^{n+1}(i, j) = (1 - \omega)f^n(i, j) + \omega\left\{\frac{1}{4}(f^n(i+1, j) + f^{n+1}(i-1, j) + f^n(i, j+1) + f^{n+1}(i, j-1)) - \frac{1}{4}u(i, j)\right\} \quad \dots (11)$$

このとき、上添え字  $n$  は反復回数を意味し、 $\omega$  は SOR 法での急速緩和係数である。この係数は 1.9 程度の値が最適値であることが多く、本研究では、 $\omega = 1.9$  として設定している。

## 6. 実験・評価

実際のイラスト画像を用いて実験を行った。また、実験の比較として、従来の Poisson 画像合成手法と、McCann 氏らの研究で提案されている Directional Blending の機能を対象とする。図 9 にある画像を実験で使用する。

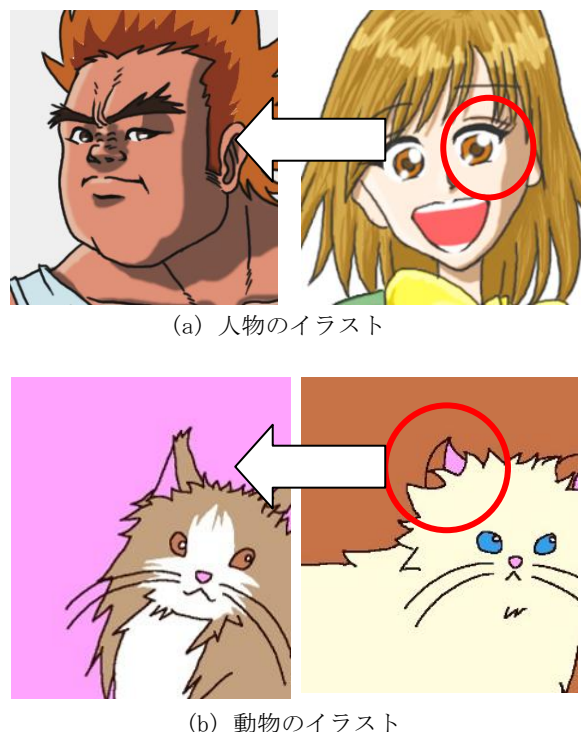


図 9. 合成に使う画像

### 6-1 制作したツール

図 10 のようなツールを制作し、実験を行った。ツールに表示されている画像のうち、左側がターゲット画像、右側がソース画像となっている。また、左上が勾配編集を行う場所、右上がソース画像から張り付ける個所を選択する場所である。そして、下段はそれぞれの元画像になっている。最後に、ツールの左側に編集時のブラシの大きさや重み、勾配強度・方向に対する閾値などの数値を設定するスライダーを設けた。この数値を変更することにより、任意の勾配に対して編集することができるようになる。

編集時に生成される重みをもとに随時 Poisson 画像合成を行うことでインタラクティブに編集することができる。

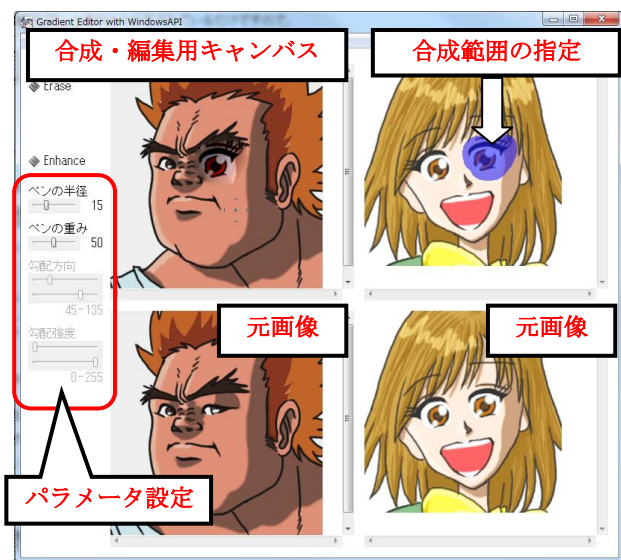


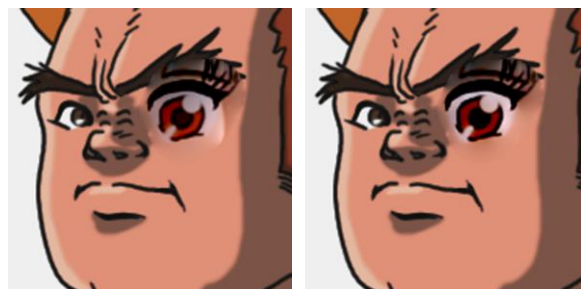
図 10. 実験ツールの全体図

### 6-2 Poisson 画像合成との比較

図 11 に図 9 の実験画像に対する結果を示す。図 9(a) では左目に対して合成を行う。図 11(a) は Poisson 画像合成で貼り付けたもので、図 11(b) は本手法を用いて勾配編集し合成したものである。図 11(a) は Poisson 画像合成を適応しただけのものである。そのため、目じりの部分に余計なエッジがあり、白目の色調が違うことも分かる。この 2 か所を本手法によって改善したものが図 11(b) である。余計なエッジを除去することに成功し、白目のコントラストもはっきりしたことが分かる。

これと同様に、図 9(b) の動物のイラストを用いて右耳に対して合成を行った。図 11(c) の合成する画像の毛の色を修正したものが図 11(d) である。ある程度の色調補正は行っているが、若干の色調のズレや輪郭線の劣化などがあり十分な結果とは言えない。現時点での本手法の適応限界と考える。

一方、渡辺氏らの研究では、Poisson 画像合成の際に生じる不具合を領域最適化によって解決する手法を提案している。この手法を用いた例が、図 12 である。図 12(d) は、領域最適化を施した後に Poisson 画像合成をしたものだが、白目の色調が違うという不具合がある。これに対し、図 11(b) では、領域最適化を行わずに合成した個所の勾配を直接編集することができるため、白目の不具合を解消しているのが分かる。



(a) Poisson 合成

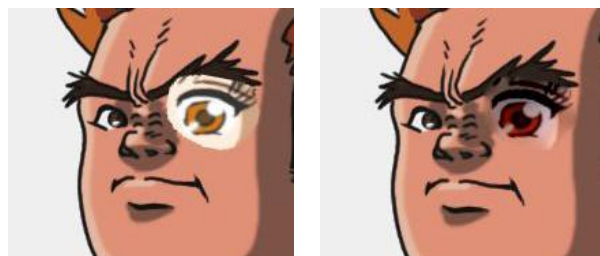
(b) 本手法



(c) Poisson 合成

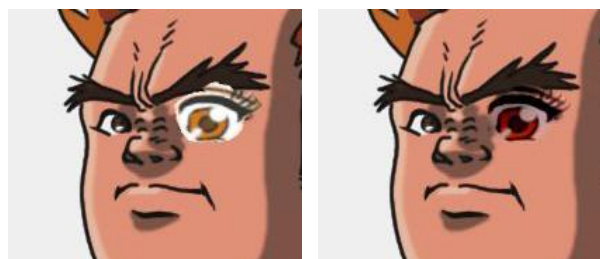
(d) 本手法

図 11. 本手法を用いた合成結果



(a) 通常合成

(b) Poisson 画像合成



(c) 領域最適化

(d) 領域最適化

+Poisson 画像合成

図 12. 領域最適化による Poisson 画像合成

### 6-3 Directional Blending との比較

McCann 氏らの研究で提案されている Directional Blending の機能を(5)式に適用する場合、次式のようにすれば良い。

$$w_g = 1 + \frac{b^x \cdot g^x + b^y \cdot g^y}{g^x \cdot g^x + g^y \cdot g^y} \quad \dots (12)$$

このとき、 $(b^x, b^y)$ はブラシストローク方向に直交な単位ベクトルである。この機能で編集した結果と本手法を比較したものが図 13 である。図 13 の上段の画像にある赤い四角で囲った部分を拡大したものが下段の画像になる。McCann 氏らの手法は本手法の勾配方向による閾値処理の場合と同じ効果を得ることができ、ブラシストローク方向と直交方向の勾配に対して編集することが可能であることが分かる。しかし、勾配強度に対しての処理が無いため、意図しない勾配にまで編集されていることが分かる。また、閾値処理を行わないことにより、余計な部分まで編集さえてしまうということがある。ほかに、ブラシの大きさを比較的小さく設定しておかないと広い範囲で勾配編集がされてしまうため、本手法よりも使いにくいことが分かる。

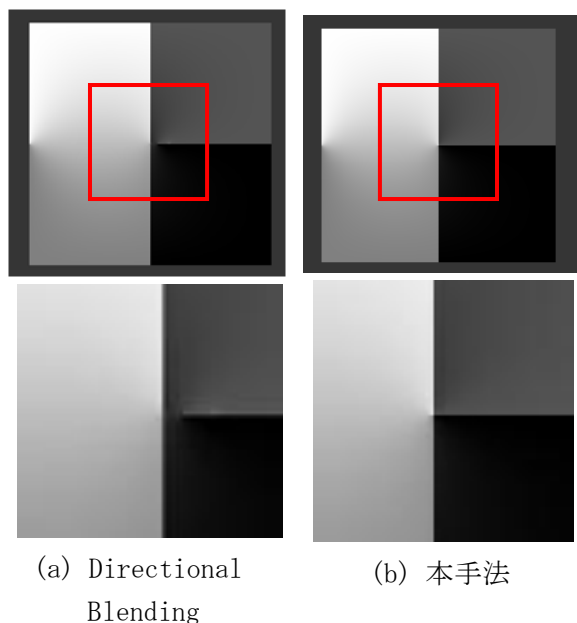


図 13. Poisson 画像合成手法との比較

#### 6-4 操作性の評価

本研究で制作したツールをもとに操作性に対する評価を行った。このとき、数人に実際に使ってもらい、操作性に関する評価をしてもらった。本研究で提案した手法では、ブラシの半径や重み、勾配の強度と方向に対する閾値で合計 6 種類のパラメータを扱うことになるため、作業に煩雑さが生まれた。それにより、ユーザが望んだ勾配を編集するためのパラメータを設定するのに時間を要した。また、異なる強度の勾配を編集する際に再度パラメータの設定をしなくてはならないという手間がある。しかし、ブ

ラシを利用したインタラクティブな編集手法であるため、直観的に容易な勾配編集ができることが分かった。

#### 7. おわりに

本研究では、勾配の強度・方向を考慮した閾値処理による勾配編集を行い、Poisson 画像合成を行う手法を提案した。これにより、Poisson 画像合成時に発生する特有の不具合を解消することができた。このとき、ブラシインタフェースによる編集を行うことにより、ユーザが直観的に編集できるようになった。また、インタラクションを確立させることにより、ユーザが微調整を行えるようにした。

しかし、本手法では、閾値処理に必要なパラメータが強度と方向に 2 つずつの計 4 つ必要となるため、数値の設定に時間がかかってしまう。今後は、閾値に必要なパラメータを自動的に選択する処理が行えることが望ましい。また、インタフェースを改善することが言える。それにより、さらに実用性の高い勾配編集ツールを開発することを課題とする。

#### 参考文献

- [1] Pérez, P., Gangnet, M. and Blake A. : Poisson Image Editing, Proc. SIGGRAPH' 03, pp. 313-318(2003)
- [2] Jia, J., Sun, J., Tang, C.-K. and Shum, H.-Y. : Drag & Drop Pasting, Proc. SIGGRAPH' 06, pp. 631-636(2006)
- [3] McCann, J. and Pollard, N. S. : Real-Time Gradient-Domain Painting, Proc. SIGGRAPH' 08, pp. 93:1-pp93:7(2008)
- [4] 宮岡伸一郎: 画像の勾配空間フィルタリング, 情報処理学会論文誌, vol. 52, No. 2 pp. 901-909(2011. 2)
- [5] 渡辺賢悟, 伊藤和弥, 近藤邦雄, 宮岡伸一郎: Poisson Image Editing を用いたキャラクターカラーシステムの開発, 芸術科学会論文誌, Vol. 9, No. 2, pp. 58-65(2010)