

パッチベースイメージキルティング探索法を用いた 2 画像の相対配置の最適化によるイメージモンタージュ法

鈴木康太¹⁾ (学生会員)

藤本忠博²⁾ (正会員)

岩手大学大学院工学研究科

Image Montage by Optimizing Relative Arrangement of Two Images using Patch-based Image Quilting Search

Kota Suzuki¹⁾(Student Member) Tadahiro Fujimoto²⁾(Member)

Graduate School of Engineering, Iwate University

h23j074@eecs.iwate-u.ac.jp

fujimoto@cis.iwate-u.ac.jp

アブストラクト

本研究では, 異なるシーンを撮影した 2 枚の入力画像の最適な相対配置を自動的に決定して自然な出力画像を合成するイメージモンタージュ法を提案する. 本手法では, 2 枚の入力画像が部分的に重なるように相対的に配置し, その重なり領域において入力画像間の色の分布が最も類似する縦横方向の画素列を最類似パスとして求める. そして, 効率的な多重解像度処理により, 全ての相対配置の中から類似度が最も高い最類似パスを持つ最適な相対配置を決定する. 最類似パスの探索は, イメージキルティングテクスチャ合成法で用いられる画素ベースの探索法をパッチベースに発展させた「パッチベースイメージキルティング探索法」によって行う. このパッチベース探索法では, 入力画像間で重なり合うパッチどうしの類似度を考え, パスに沿った連続するパッチ群の類似度が最も高いものを最類似パスとして得る. パッチ間の類似度の評価にはカラーヒストグラムを用いる. 最適な相対配置の決定後は, 2 枚の入力画像をつなぐ「境界領域」と出力画像上に複製する各入力画像上の「複製領域」を縦横方向の最類似パスから決定する. そして, 境界領域に対するフェザリングとテクスチャ合成, あるいは, ポアソン画像合成によって自然な出力画像を得る. なお, 出力画像上に残したい各入力画像上の「保持領域」の指定も可能である.

Abstract

In this research, we propose an image montage method to synthesize a natural-looking output image by automatically determining the optimal relative arrangement of two input images of different scenes to montage. In this method, the two input images are relatively arranged so as to overlap each other partially. The overlapping region is searched for the most similar paths in both of vertical and horizontal directions, each of which is the chain of pixels with the most similar color distribution between the two input images. An efficient multiresolution method determines the optimal relative arrangement that has the most similar paths with the highest similarity among all relative arrangements. The most similar path is found by the patch-based image quilting search method, which we have developed based on the pixel-based search method used in the image quilting texture synthesis. This patch-based search method evaluates the similarity between two patches overlapping each other, which come from the two input images respectively. The method finds the most similar path that has the chain of patches with the highest similarity along the path. The similarity between two patches is evaluated using their color histograms. After determining the optimal relative arrangement, the “boundary region” and “copy regions” are determined using the most similar paths of both directions. The two input images are stitched together along the boundary region. The copy region copied from each input image is pasted on the output image as it is. The output image is completed by applying feathering and texture synthesis, or Poisson image editing to the boundary region. A “preserved region” can be assigned in each input image in order to include the region on the output image.

1. はじめに

コンピュータグラフィックスやコンピュータビジョンの分野で、元になる入力画像から新たな出力画像を合成するイメージエディットと呼ばれる様々な画像合成技術が盛んに研究されている。これらの技術の一つとして、複数の入力画像を組み合わせて1枚の自然な出力画像を合成するイメージモンタージュ (image montage) と呼ばれる技術がある。通常、各入力画像を重ね合わせる相対的な配置はユーザによって決定されることが多い。しかし、例えば、複数の異なる風景の入力画像から1枚の自然な風景の出力画像を合成したい場合などでは、自動的に最適な相対配置を決定することができれば、その有用性は高い。そこで、本研究では、2枚の入力画像上で互いに似た雰囲気、すなわち、類似する画素色のパターンを持つ部分どうしが重なり合うような最適な相対配置を自動的に決定し、その部分を境界としてつなげることで、違和感のない自然な出力画像を合成する手法を提案する。本手法では、テキストチャ合成の一手法であるイメージキルティング (Image Quilting) 法[1]における画素ベースの最類似パス探索法をパッチベースに発展させた「パッチベースイメージキルティング探索法」を提案し、2枚の入力画像の最適な相対配置、ならびに、それらをつなぎ合わせる境界とする最類似パスを求める。2枚の入力画像のパッチ間の類似度の評価には Lab 表色系における a, b 値によるカラーヒストグラムを用いる。最適な相対配置と最類似パスの決定後は、出力画像を合成するため、まず、入力画像が配置されない出力画像の角の空白領域にテキストチャ合成[1]を行う。その後、最類似パスから得られる境界領域に対する合成処理によって出力画像を完成させる。この合成処理として2つの方法の適用を試みる。一つ目の方法では、まず、境界領域に対して、2枚の画像の重なり領域上で両側の境界からの距離に応じた重み付けを行うブレンド手法であるフェザリング[2]を行う。そして、出力画像がより自然に見えるように、ブレンドした境界領域が入力画像上に実在する画素色のパターンで置き換わるように反復法によるテキストチャ合成[3]を適用し、出力画像を完成させる。二つ目の方法では、境界領域に対して2枚の入力画像を用いたポアソン画像合成 (Poisson Image Editing) [4]を適用し、入力画像どうしの自然な融合を試みる。また、本手法では、各入力画像内で意図的に出力画像上にそのまま複製したい部分を保持領域として指定することも可能である。

以下、2節で関連研究を紹介し、3節でイメージキルティング法について説明する。そして、4節で本研究の提案手法について説明し、5節で実験結果を示し、6節で結論を述べる。

2. 関連研究

複数の入力画像から1枚の出力画像を合成する技術として、これまでに様々な手法が提案されている。

文献[5]のモンタージュ法では、出力画像上での配置を考慮して手動で重ねられた複数の入力画像に対して、出力画像上に合成したい各入力画像上の領域をインタラクティブにストローク

描画で指定することで出力画像が合成される。異なる入力画像を指定する複数のストロークから、グラフカット最適化により出力画像上での各入力画像の領域間の最適な境界が決定される。文献[6]のシステムでは、出力画像上に合成したい複数の物体について、それらのカテゴリーのラベル (例えば、「空 (sky)」, 「木 (tree)」, 「道路 (road)」など) をインタラクティブにキャンバス上に配置すると、その配置を実現するような各物体を含む入力画像がデータベースから探索され、グラフカット最適化により各入力画像上で物体が含まれる領域の適切な境界が決定され、出力画像が合成される。ラベルの配置だけでなく、キャンバス上のペイント描画で物体を配置する領域を指定することや、どのカテゴリーにも属さない特定の物体を手動で配置することも可能である。しかし、出力画像上で物体領域間の境界が自然に見えるような処理は行われない。文献[7]は、グラフカット最適化を利用したテキストチャ合成法を提案しており、その応用として、2枚の入力画像を適切な境界でつなぎ合わせて自然な1枚の出力画像に合成できることを示している。適切な境界は自動で得られるが、2枚の入力画像の相対配置は手動で行う必要がある。

イメージペースティング (image pasting) と呼ばれる技術は、2枚の入力画像 (前景画像, 背景画像) を用い、前景画像から切り抜いた前景領域を背景画像上に手動で配置して出力画像を合成する。文献[4]と[8]は、ポアソン方程式の解法を利用したペースティング法を提案しており、前景領域と背景画像の明暗の違いや境界が目立たない自然な出力画像を合成する。

イメージコンプリーション (image completion) と呼ばれる技術は、不要な物体の削除などで生じた入力画像内の空白領域を違和感なく自然に埋める。入力画像内の残された領域の画素色を使って埋めることもあるが、他の画像を用いることもある。その場合、入力画像と他の画像から出力画像を合成するモンタージュ技術であるとも考えられる。文献[9]のコンプリーション法は、入力画像上でマスキングにより指定した合成希望領域に対して、テキストで提示した属性 (例えば、風景画像上の「空 (sky)」の部分が合成希望領域であれば、「晴れ (clear)」, 「曇り (cloudy)」, 「夕焼け (sunset)」など) を持つ適切な合成用画像がデータベースから探索され、選ばれた合成用画像を用いたテキストチャ合成によって合成希望領域を埋めた出力画像を生成する。文献[10]のコンプリーション法は入力画像として風景画像を対象とし、不要物体の削除で生じた空白領域を自然に埋めるための最適な合成用画像を大規模データベースから探索する。そして、境界が目立たない最適な合成領域を合成用画像内で自動的に決定して出力画像を生成する。

簡略化 (summarization) 技術は、画像や動画に含まれる重要な視覚情報を残し、重要でない冗長な部分を削減することで、全体的に画面サイズや再生時間の短縮を図る。文献[11]が提案する時空間ビデオモンタージュ法は、グラフカット最適化により動画の簡略化を行う。入力動画中の視覚的に重要な情報を抽出し、なるべく小さな時空間領域 (短時間で小さな画面サイズ) に収める。しかし、同一時刻の画面内に全く異なるシーンを埋め込むなど、1つのシーンとして視覚的に自然に見せることが

目的ではない。文献[12]は入力画像の特徴的な部分が残されているかを測る「完全性」と出力画像が自然に見えるかを測る「一貫性」を評価する双方向類似度 (bidirectional similarity) を利用して画像や動画を簡略化する手法を提案している。その応用として、複数の入力画像を1枚の自然な出力画像に合成するモンタージュが可能であることを示している。ただし、出力画像上での最終的な配置を考慮して、各入力画像を手動で適切に初期配置する必要がある。文献[13]は、入力画像上で重要度の低い部分を縦横方向の画素列として探索して削除することで、重要度の高い部分を維持しながら、画像の大きさを変更する手法を提案している。しかし、複数の入力画像を合成するモンタージュ法ではない。

文献[14]は、画像の大域的な特徴を定量化する GIST 特徴量を利用して、部分的に類似する特徴を持つ複数の入力画像を逐次的にデータベースから探索して合成していく手法を提案している。ただし、類似性を評価する入力画像上の領域は事前に決めておく必要がある。

文献[15]のイメージメルディング (Image Merging) 法は、パッチベースの処理により、2つの入力テクスチャを自然に融合し、一方から他方へ徐々に遷移する出力テクスチャを合成する。その応用として、2つの入力画像を用いて、一方の画像内の背景を他方の背景に変えることなどが可能であることを示している。

文献[16]と[17]は、グラフカット最適化により、複数の入力画像中の重要な部分を自動的に判定して切り出し、パッチワークのように1枚の出力画像内になるべくコンパクトに収める相対配置を決定する手法を提案している。しかし、得られる出力画像は異なるシーンの断片を並べたパッチワークであり、一つのシーンとして自然に見える出力画像を得ることが目的ではない。

著者らの知る限りでは、これまでに提案されてきた複数の入力画像から1枚の出力画像を合成する手法のうち、境界部分でのつながり等を考慮して違和感のない自然な出力画像を得ることを目的とするものは、入力画像の相対配置を手動で行うことを必要とする手法がほとんどである。また、逆に、適切な相対配置を自動的に決定する手法では、出力画像上での境界部分の自然なつながりを実現しないものや、1つのシーンとして自然に見えることを目的とはしないものが多い。そこで、本研究では、2枚の入力画像について、境界部分の自然なつながりを実現するような最適な相対配置を自動的に決定し、1つのシーンとして自然に見える出力画像を合成するイメージモンタージュ法の提案を目的とする。

3. イメージキルティング法

イメージキルティング法[1]はテクスチャ合成の一手法であり、入力テクスチャから切り出した複数のパッチを境界部分が重なるように貼り合わせて出力テクスチャを合成するパッチベース手法である。複数のパッチを一つずつ逐次的に出力テクスチャ上に合成していくが、各パッチの合成においては、既に合成済みの領域との重なり部分が類似するパッチを入力テクスチャ上で探索し、さらに、重なり部分で最も類似性が高い画素列

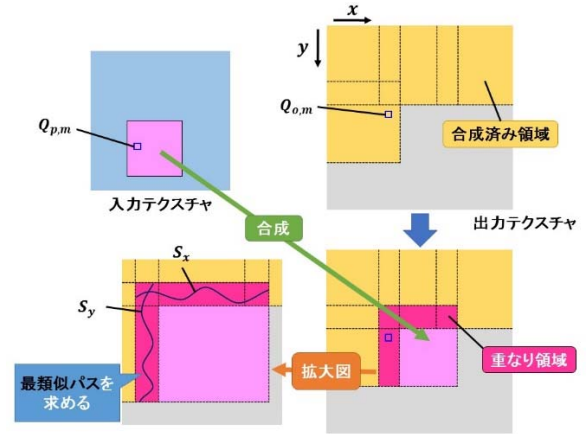


図1. イメージキルティング法

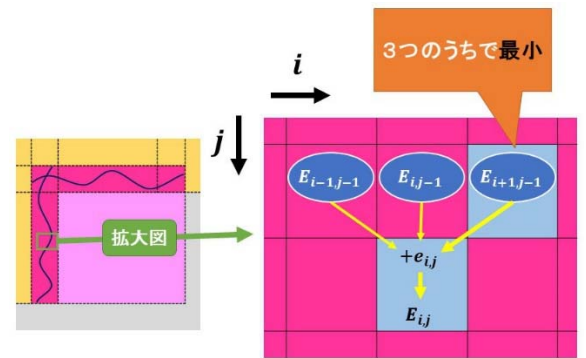


図2. 最類似パスの探索

を求めて合成の境界とすることで、つなぎ目を目立たなくする。

図1に示すように、出力テクスチャ上で走査線順 (x, y 方向のそれぞれでパッチの幅から重なり部分の幅を除いた間隔ごと) の位置でパッチを合成していく。一つのパッチの合成においては、まず、出力テクスチャの合成済み領域と重なり部分の色パターンが類似する入力テクスチャ上のパッチを探索する。重なり部分内で同じ位置で重なる M 組の合成済み領域の画素 $Q_{o,m}$ とパッチの画素 $Q_{p,m}$ について、それぞれの色値を $C_{o,m} = [R_{o,m}, G_{o,m}, B_{o,m}]$ と $C_{p,m} = [R_{p,m}, G_{p,m}, B_{p,m}]$ とする。色パターンの類似性は、それらの色値の差の二乗の和 (Sum of Squared Difference: SSD) による相違度 e を用いて評価する。

$$e = \sum_{m=1}^M |C_{o,m} - C_{p,m}|^2$$

$$= \sum_{m=1}^M \left\{ (R_{o,m} - R_{p,m})^2 + (G_{o,m} - G_{p,m})^2 + (B_{o,m} - B_{p,m})^2 \right\} \quad (1)$$

相違度 e が小さいほど類似性が高い。相違度 e が閾値以下となるパッチを候補とし、その中から合成に用いる一つのパッチ P_{opt} をランダムに選択する。

続いて、図1に示すように、重なり部分内にある合成済み領域とパッチ P_{opt} の重なり合う画素の列のうち、色パターンが

最も類似する縦方向の画素列 S_y と横方向の画素列 S_x を求める。以下、画素列をパスと呼ぶ。図2に縦方向の最類似パスの探索の方法を示す。重なり部分内の画素位置 (i, j) における相違度を $e_{i,j}$ とし、次式によって累積相違度 $E_{i,j}$ を最上部の画素から下向きに求めていく。

$$\begin{aligned} E_{i,1} &= e_{i,1}, \\ E_{i,j} &= e_{i,j} + \min(E_{i-1,j-1}, E_{i,j-1}, E_{i+1,j-1}) \end{aligned} \quad (2)$$

その後、最下部で累積相違度 $E_{i,j}$ が最小の画素 Q_{min} を選び、画素 Q_{min} に最小の累積相違度を与えた画素列を上向きに辿り、最類似パスを得る。横方向の最類似パスも同様の方法で求める。最終的に、2方向の最類似パスで分けられるパッチ P_{opt} の4つの領域のうち、右下の領域だけを合成する。

4. 最適な相対配置によるイメージモンタージュ法

4.1 概要

本研究では、異なるシーンを撮影した2枚の入力画像を用いて、それらの最適な相対配置を自動的に決定し、一つのシーンとして違和感なく自然に見える出力画像を合成するイメージモンタージュ法を提案する。本手法では、2枚の入力画像上で似た雰囲気、すなわち、類似する画素色のパターンを持つ部分を探し、その部分を境界としてつなげて出力画像を合成することを方針とする。そのため、2枚の入力画像が部分的に重なるように相対的に配置し、その重なり領域において互いに類似するパターンを持つ部分どうしが重なり合っているかどうかを調べる。そして、様々な相対配置のうち、最も類似するパターンの重なり部分が境界として生じる配置を求め、その境界に沿って2枚の入力画像をつなげて出力画像を合成する。

上記の方針を実現するため、2枚の入力画像の重なり領域において3節で述べたイメージキルティング法の最類似パスの探索法を利用する。しかし、従来の画素ベースの探索法により1画素の幅で入力画像間の類似度を評価しただけでは、類似するパターンを持つ部分であるかを十分には評価できない。そこで、より広い画素幅で適切な評価を行うため、縦横方向に数画素幅を持つパッチを評価の単位とし、従来の画素ベースの探索法をパッチベースに発展させた「パッチベースイメージキルティング探索法」を提案する。そして、縦方向と横方向のそれぞれで、重なり領域内で連続するパッチ群からなるパスについて、入力画像間で同じ位置で重なり合うパッチどうしの類似度が最も高くなるような最類似パスが得られる相対配置を決定する。この際、2枚の入力画像を重ねて1画素ずつずらしながら、全ての相対配置について最類似パスを求めて類似度を評価するのは多くの計算時間を要する。そこで、入力画像を多重解像度化することで効率的に最適な相対配置を決定する。この多重解像度処理は、計算の効率化だけではなく、より良好な相対配置を求めるためにも有効である。

異なるシーンの入力画像では、パスの全体に渡って入力画像間で同じ位置で重なり合う画素どうしが類似色を持つことは稀

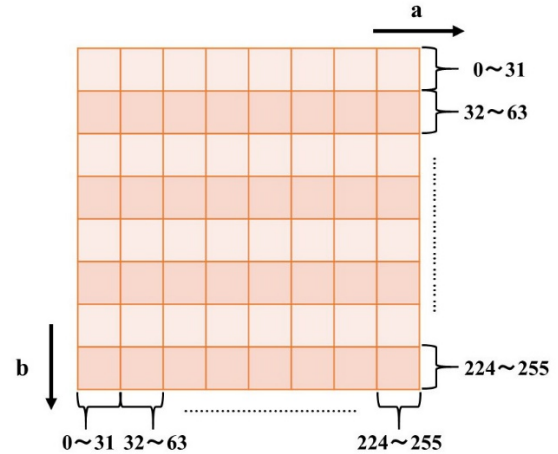


図3. パッチ間の類似度評価のためのカラーヒストグラム

有である。そこで、上述したように、2枚の入力画像上で類似する画素色のパターンを持つ重なり部分が境界として生じる相対配置を得るという方針からも、パスに沿って色の分布が似ていることを類似度の指標とし、パッチ間の類似度の評価には画素色の分布を定量化するカラーヒストグラムを用いる。

相対配置の決定後は、縦横方向の最類似パス、ならびに、その近傍領域から、2枚の入力画像をつなげる「境界領域」を決定する。そして、それ以外の領域から、出力画像上にそのまま複製する各入力画像上の「複製領域」を決定する。その後、矩形状の出力画像を完成させるために、まず、入力画像が配置されない出力画像の角の空白領域にテクスチャ合成[1]により適切な画素色を与える。そして、境界領域に対する合成処理によって出力画像を完成させる。この合成処理として2つの方法の適用を試みる。一つ目の方法では、まず、境界領域の両側の各入力画像の複製領域どうしが自然に調和するように、フェザリング[2]によるブレンド処理を行う。そして、ブレンドした境界領域が自然に見えるように、その領域を入力画像上の実在する画素色のパターンで置き換えるための反復法によるテクスチャ合成[3]を適用する。二つ目の方法では、境界領域に対してポアソン画像合成[4]を適用することで、境界領域内で重なる2枚の入力画像上の特徴的なパターンを融合して自然に見えるようにする。また、本手法では、入力画像内で意図的に複製領域としたい部分を「保持領域」として事前に指定することで、保持領域を制約として相対配置を決定することも可能とする。

4.2 カラーヒストグラムによるパッチの類似度評価

2枚の入力画像のパッチ間の類似度を評価するカラーヒストグラムは、以下のように求める。まず、2つのパッチ内の各画素が持つ色値を Lab 表色系で表現する。明暗よりも色の種類を優先するため、明度 L は用いず、色相と彩度をあらわす a と b ($0 \leq a, b \leq 255$) を用いる。そして、 a と b のそれぞれの値が取る区間 $[0, 255]$ を R 分割してできる R^2 個のビンにパッチ内の各画素の色値 (a, b) を登録した2次元のカラーヒストグラムを求める (図3 参照)。2つのパッチ P_m ($m = 1, 2$) のヒストグラムのビン i ($1 \leq i \leq R^2$) に登録された画素数を

$H_m[i]$ とし、以下のように正規化を行う.

$$\overline{H}_m[i] = \frac{H_m[i]}{\sum_{i=1}^{R^2} H_m[i]} \quad (3)$$

そして、2つのヒストグラムの類似度を次式のヒストグラム交差法 (histogram intersection) [18]により求める.

$$d(\overline{H}_1, \overline{H}_2) = \sum_{i=1}^{R^2} \min(\overline{H}_1[i], \overline{H}_2[i]) \quad (4)$$

類似度 d は $0 \leq d \leq 1$ の値となり、2つのヒストグラムが完全に等しければ1となり、類似度が低くなるにつれて0に近づく. 本手法では、この類似度 d を2つのパッチの類似度の評価に用いる.

4.3 最適な相対配置の決定

最適な相対配置を決定する方法としては、まず、候補とする各相対配置のもとで、2枚の入力画像の重なり領域内でパッチを単位とした最類似パスを求め、その類似度を計算する. そして、類似度が最も高い最類似パスを持つ相対配置を最適なものと決定する. このとき、候補とする相対配置として、2枚の入力画像を重ねて1画素ずつずらした全ての相対配置を候補とするのは多くの計算時間を要する. そこで、入力画像を多重解像度化することで、以下のアルゴリズムにより効率的に最適な相対配置を決定する.

ステップ1: 2枚の入力画像 I_n ($n = 1, 2$) のそれぞれについて、元の入力画像 I_n を I_n^0 として、縦横方向の解像度を $1/2$ ずつ小さくした画像 I_n^k ($k = 1, 2, \dots, K$) を生成する. 縦横方向のそれぞれで I_n^k の画素数は I_n^0 の $(1/2)^k$ 倍になる.

ステップ2: 最低解像度の入力画像 I_1^K と I_2^K について、縦横方向に1画素ずつずらした全ての相対配置に対して、最類似パスとその類似度を求める.

ステップ3: ステップ2の全ての相対配置のうち、事前に決めておいた上位選択数 $T \geq 1$ に対して、類似度が高いほうから上位 T 番目までに入る相対配置を求め、それらを A_t^K ($t = 1, 2, \dots, T$) とする.

ステップ4: 以下の処理を $k = K, K-1, \dots, 2, 1$ の順で繰り返す.

ステップ4-1: 入力画像 I_1^k と I_2^k に対して求めた T 個の相対配置 A_t^K ($t = 1, 2, \dots, T$) のそれぞれについて、その配置をそのままにしたものを解像度が1段階だけ高い入力画像 I_1^{k-1} と I_2^{k-1} の基本相対配置 $A_{t(0,0)}^{k-1}$ とする. これは A_t^K の縦横方向の画素座標を2倍にすることで得られる.

ステップ4-2: 基本相対配置 $A_{t(0,0)}^{k-1}$ から縦横方向に正負1画素の範囲でずらした9個の相対配置 $A_{t(i,j)}^{k-1}$ ($i = -1, 0, 1, j = -1, 0, 1$) に対して、それぞれ、最類似パスとその類似度を求める.

ステップ4-3: 合計 $9T$ 個の相対配置 $A_{t(i,j)}^{k-1}$ のうち、最類似パスの類似度が高いほうから上位 T 番目までに入る相対配置

を求め、それらを A_t^{k-1} ($t = 1, 2, \dots, T$) とする.

ステップ5: ステップ4の終了時には、最高解像度の元の入力画像 I_1^0 と I_2^0 について、最類似パスの類似度が高いほうから上位 T 番目までに入る相対配置 A_t^0 ($t = 1, 2, \dots, T$) が得られる. 最終的に、上位1番目の相対配置 A_1^0 を最適な相対配置とする.

多重解像度処理を行わず、元の入力画像の最高解像度のみによる単一解像度処理で1画素ずつずらしながら最適な相対配置を求めた場合、パッチの大きさ等によっては、不適切な相対配置に対して偶発的に類似度の値が高くなるなど、2枚の入力画像から1枚の自然な出力画像を合成するという目的に対して、必ずしも良好な相対配置が得られない可能性もある. 一方、上記の多重解像度処理によるアルゴリズムは、はじめに低解像度で入力画像全体に関しておおまかに類似している部分が重なるような相対配置を求め、段階的に高解像度化することで徐々に適切な相対配置に近づけていく. この多重解像度処理は、計算の効率化のためだけではなく、より良好な相対配置を得るためにも有効であると考えられる.

4.4 パッチベースイメージキルティング探索法

4.3節で述べた最適な相対配置を決定するアルゴリズムにおいて、各解像度レベルにおける各相対配置のもとで、2枚の入力画像の重なり領域内でパッチを単位とした最類似パスを求める方法として、パッチベースイメージキルティング探索法を提案する. これは、3節で述べた従来の画素ベースの探索法をパッチベースに発展させたものである. パッチ間の類似度の評価には、4.2節で述べたカラーヒストグラムの類似度 d を用いる.

パッチベースイメージキルティング探索法の具体的なアルゴリズムの大枠は3節で述べた従来法と同じであり、2枚の入力画像の重なり領域内において、類似度が最大となる縦横方向の画素列 S_y と S_x を最類似パスとして求める. ただし、従来法では重なり合う2つの画素間の類似度を式(1)の色値を用いた相違度 e により評価するが、提案するパッチベースの探索法では各画素を中心としたパッチを用い、その2つのパッチ間の類似度を式(4)のカラーヒストグラムを用いた類似度 d により評価する. 縦方向と横方向のそれぞれで、パス上の全ての画素位置におけるパッチ間の類似度 d の総和を画素位置の数で割り、1つの画素位置あたりの平均値としたものをパスの類似度とする. そして、その相対配置における最類似パスの類似度としては、縦方向と横方向の最類似パスの類似度を加算した値 (4.2節から $0 \leq d \leq 1$ のため、加算した値の最大値は2である) を用いる.

なお、上記の探索法で得られる縦方向と横方向の最類似パスは、それぞれ、画素列 S_y と S_x を中心軸としたパッチ幅を持つ領域として扱う.

上記のパッチベースの探索法では、パス上の画素位置ごとにパッチ間の類似度 d を計算するため、その画素位置を中心とした2つのパッチのカラーヒストグラムを求める必要がある. パッチごとに各ビンに登録されている画素数を数え直すのは大

きな処理時間を要する。そこで、Summed Area Table[19]や Integral Image[20]に基づき、以下の方法で効率的に各ピンの登録画素数を得る。まず、ピン i ($1 \leq i \leq R^2$) ごとに入力画像の解像度 R_x, R_y と同じ大きさのメモリ領域 $B[i][y][x]$, $C[i][y][x]$, $D[i][y][x]$ ($0 \leq x < R_x, 0 \leq y < R_y$) を用意する。そして、入力画像の画素 (x, y) の色値 (a, b) がピン i に登録されれば $B[i][y][x] = 1$, 登録されなければ $B[i][y][x] = 0$ とする。続いて、次のように $B[i][y][x]$ の値を x, y 方向に累積して累積登録画素数 $D[i][y][x]$ を求める。

$$D[i][y][x] = \sum_{0 \leq u \leq x, 0 \leq v \leq y} B[i][v][u] \quad (5)$$

これは以下の処理で効率的に求められる。

```
for(y = 0, 1, ..., R_y - 1) {
    C[i][y][0] = B[i][y][0];
    for(x = 1, 2, ..., R_x - 1)
        { C[i][y][x] = C[i][y][x - 1] + B[i][y][x]; }
}
for(x = 0, 1, ..., R_x - 1) {
    D[i][0][x] = C[i][0][x];
    for(y = 1, 2, ..., R_y - 1)
        { D[i][y][x] = D[i][y - 1][x] + C[i][y][x]; }
}
```

以上を前処理とし、累積登録画素数 $D[i][y][x]$ を記憶しておくことで、入力画像の画素範囲 $x_0 \leq x < x_1, y_0 \leq y < y_1$ のパッチ P_m に関するピン i の登録画素数は次式で容易に得ることができる。

$$H_m[i] = D[i][y_1][x_1] - D[i][y_0 - 1][x_1] - D[i][y_1][x_0 - 1] + D[i][y_0 - 1][x_0 - 1] \quad (6)$$

パッチベースイメージキルティング探索法で用いるパッチの大きさは、中心の画素から上下左右の方向に任意の画素数とすることが可能である（パッチの一边の画素数は奇数となる）。ここで、4.3 節の多重解像度処理によるアルゴリズムでの利用を考えた場合、得られる最適な相対配置をより良好なものとするため、パッチサイズ変化率 $0 < G < 1$ を与えることで、解像度レベルごとにパッチの大きさを可変とする。4.3 節のアルゴリズムの記述に従い、元の入力画像の解像度レベル $k = 0$ におけるパッチの一边の画素数（初期パッチサイズ）を S_0 画素とした場合、解像度レベル $k = 1, 2, \dots, K$ におけるパッチサイズは $S_k = S_0 G^k$ 画素となる。入力画像の解像度は $1/2$ ずつ減少するため、変化率が $G = 1/2$ の場合には、解像度の減少と同じ割合でパッチが小さくなり、全ての解像度レベルで一つのパッチが覆う入力画像内の範囲は相対的に等しくなる。変化率が $G > 1/2$ の場合には、解像度の減少に伴って一つのパッチが覆う入力画像内の範囲は相対的に大きくなっていく。なお、上記による S_k の計算値は一般には奇数にならないため、例えば、計算値以下の最大の奇数とするなど、適切な変換によって

奇数とする。

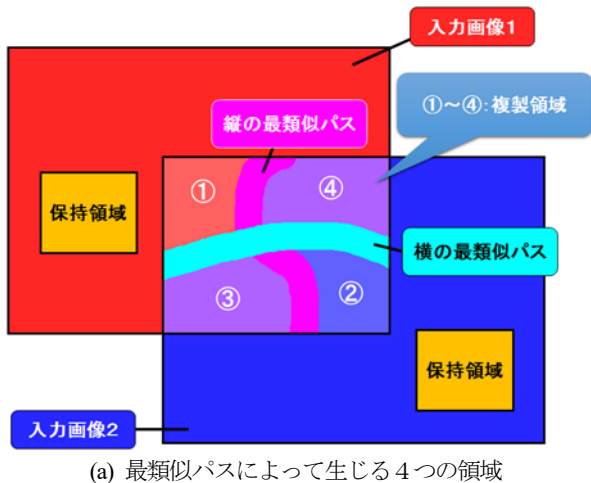
4.5 保持領域の指定

本手法では、各入力画像内で意図的に出力画像上にそのまま複製したい部分を保持領域として指定することができる。その場合、保持領域を制約として最適な相対配置を決定する。具体的には、後述のように、最類似パスがフェザリングとテクスチャ合成、あるいは、ボアソン画像合成により元の画像の内容が失われてしまう境界領域になり得ることから、保持領域に重なるパッチが含まれるパスを最類似パスの候補から除外することで実現される。

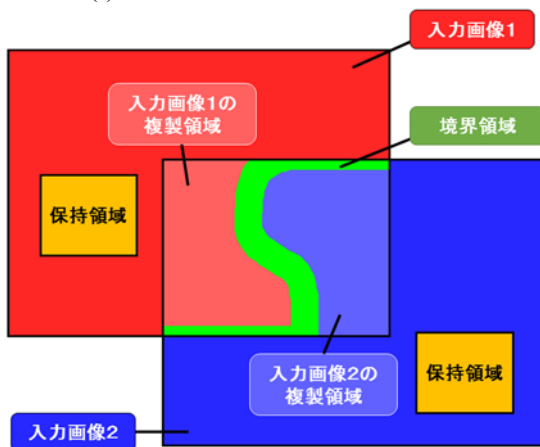
4.6 境界領域と複製領域の決定

2枚の入力画像の最適な相対配置を決定した後、縦横方向の最類似パスを境界として2枚の入力画像をつなげて出力画像を合成する。その際、最類似パス上のパッチの類似度から、実際に2枚の入力画像をつなげる境界領域を決定する。そして、その境界領域から、それぞれの入力画像内で出力画像上にそのまま複製する複製領域を決定する。このとき、相対配置の状況から、2枚の入力画像の重なり領域内では、どちらの入力画像の複製領域とすべきかが明白である領域があるため、これを考慮して境界領域の決定を行う。

図4に例を示す。図4(a)のように入力画像1と入力画像2の最適な相対配置が決定した場合、縦横方向の最類似パスによって2枚の入力画像が重なる4つの領域①～④ができる。これらのうち、各入力画像との位置関係から、領域①は入力画像1の複製領域、領域②は入力画像2の複製領域に自動的に決定される。一方、領域③と④はどちらの入力画像の複製領域にもなることができる。ここで、縦方向と横方向の最類似パスは、それぞれ、それらが交差する箇所まで2個のサブパスに分断され、合計で4個のサブパスができる。この4個のサブパスから、上記の領域①～④の複製領域に関する条件を満たすように、2個のサブパスを選択して組み合わせたものを境界領域とする。具体的には、パッチ間の類似度 d を利用して、各サブパス上の1つの画素位置あたりの類似度の平均値を求め、その値が大きい2個のサブパスを選択することとする。このサブパスの選択により、同時に複製領域も決定される。図4(b)は、縦方向の最類似パスに属する2個のサブパスが選択された場合である。結果として、領域③は入力画像1の複製領域、領域④は入力画像2の複製領域となる。このとき、入力画像1と入力画像2の端から端までを結ぶ境界領域を構成するため、選択された2個のサブパスに沿った領域だけでなく、それにつながる各画像の端部に沿った領域も用いることで、図中の緑色の領域を境界領域として決定する。上記のサブパスに沿った領域はパッチ幅を持つことを基本とするが、実際には、より自然な出力画像を得るため、パッチ幅から外側に数画素分の拡張を行った領域を境界領域とする。また、各画像の端部に沿った境界領域には任意の数画素分の幅を持たせる。なお、2枚の入力画像の重なり領域以外、すなわち、重ならない領域は、それぞれの入力画像の複製領域とする。保持領域は各入力画像の複製領域に含まれる。



(a) 最類似パスによって生じる4つの領域



(b) 境界領域と複製領域の決定

図4. 境界領域, 複製領域, 保持領域

4.7 出力画像の合成

最適な相対配置の決定のためのパッチ間の類似度の評価においては、明暗よりも色の種類を優先するため、明度 L は用いず、色相と彩度をあらわす a と b だけを用いている。そのため、出力画像の合成の前に、2枚の入力画像が違和感なく合成されるように各画像の明度 L を調整する。

次に、矩形形の出力画像を完成させるために、出力画像の角の部分で入力画像が配置されない空白領域にテクスチャ合成により適切な画素色を与える。このテクスチャ合成には、パッチベースのイメージキルティング法[1]を用いた。最後に、4.6節の方法で決定した境界領域に対する合成処理によって出力画像を完成させる。この合成処理として以下の2つの方法の適用を試みる。

一つ目の方法では、まず、境界領域に対して、その両側の各入力画像の複製領域どうしが自然に調和するように、フェザリング[2]によるブレンド処理を行う。フェザリングとは2枚の画像を重ね合わせてブレンドする一手法であり、重なり領域の両側の境界からの距離に応じた重み付けを行う。各画像に対して、近いほうの境界上では1、遠いほうの境界上では0となるように線形に変化する重みを用いる。しかし、このシンプルなブレンド処理だけでは異なるシーンの入力画像どうしを自然に調和

させることは難しい。そこで、そのブレンド済みの境界領域をより自然に見せるため、その画素色のパターンを入力画像上の実在するパターンで置き換えるように、テクスチャ合成により入力画像上の適切な画素色を合成する。これを実現するため、ブレンドによる画素色を初期状態とし、入力画像上で類似する近隣画素群 (neighbor) の探索と出力画像への合成を反復して大域的に境界領域全体を収束させるテクスチャ合成法[3]を用いる。

二つ目の方法では、境界領域に対してポアソン画像合成[4]を適用する。それぞれの入力画像の複製領域となる境界領域の外周の画素色を境界条件とし、境界領域の内部で入力画像の画素色の微分を積分することで、境界領域の外部から内部に向けて2枚の入力画像の自然な融合がなされる。境界領域内では、画素ごとに2枚の入力画像の微分のうちで絶対値が大きくなるほうを利用することで、より特徴的なパターンが出力画像上に自然に現れることが期待される。

5. 実験

実験結果を以下に示す。実験用の画像には文献[21-25]に示すウェブサイトのフリー画像を用いた。なお、特に明記していない場合、デフォルトの値として、ヒストグラムのビン分割数 $R = 8$ 、多重解像度処理における解像度レベル数 $K = 4$ 、上位選択数 $T = 5$ 、初期パッチサイズ (元の入力画像の解像度レベル $k = 0$ でのパッチの一辺の画素数) $S_0 = 25$ 、パッチサイズ変化率 $G = 0.5$ とした。また、5.1節から5.9節に関する実験結果である図5から図13では、4.7節で述べた合成処理の方法としてフェザリングとテクスチャ合成を用いた出力画像を示す。それ以降の実験結果の図では、フェザリングとテクスチャ合成、ならびに、ポアソン画像合成の両方による出力画像を示す。

5.1 初期パッチサイズの影響

初期パッチサイズ S_0 の影響を調べた実験の結果を図5に示す。なお、この実験では、デフォルトのパッチサイズ変化率 $G = 0.5$ を用いるため、多重解像度処理における解像度の減少と同じ割合でパッチサイズが減少する。入力画像1と入力画像2に対して、本手法で得られた最適な相対配置と最類似パス、境界領域、出力画像を示している。入力画像2については、出力画像の合成には明度 L を調整したもの (図中の「明度 L 調整後」) を用いている。図中で、入力画像1、入力画像2、ならびに、「最適な相対配置と最類似パス」の上に描かれたピンク色の矩形は、出力画像上にそのまま複製するために指定した保持領域である。「最適な相対配置と最類似パス」の上に描かれた赤と青の帯は、縦方向 (赤) と横方向 (青) について得られた最類似パスをパッチ幅で表示したものであり、最類似パスが重なるように2枚の入力画像を重ねたものが「最適な相対配置と最類似パス」の画像である。「境界領域」の上に描かれた帯は、4.6節で述べた方法で縦横方向の最類似パスから選択された境界領域であり、入力画像の端部に沿った境界領域は省略されている。出力画像は、4.7節で述べたように、「境界領域」の画像上の空



図 5. 初期パッチサイズの影響

白領域をパッチベースのテクスチャ合成，境界領域を合成処理の一つ目の方法として述べたフェザリングと反復法によるテクスチャ合成によって補完して生成されたものである．(d)，(e)，(f)は，初期パッチサイズ $S_0 = 5$ ，ヒストグラムのビンの分割数 $R = 4$ のときの結果である．出力画像(f)では，中央上部の建物の一部が欠落しているなど，良好な結果は得られていない．一方，(g)，(h)，(i)は， $S_0 = 25$ ， $R = 8$ のときの結果であり，前者に比べて良好な結果が得られている．この原因としては，前者の場合にはパッチサイズが小さすぎて色の分布の特徴が十分にとらえられず，一方，後者の場合には十分にとらえられたと

ということが考えられる．一般には，最適なパッチサイズは入力画像の特徴に依存するものと考えられる．さらに，前者では，ヒストグラムのビンの個数が $R^2 = 4^2 = 16$ であるのに対して，最高解像度レベル $k = 0$ でもパッチ内の画素数は $S_0^2 = 5^2 = 25$ だけであり，平均で一つのビンあたり $S_0^2/R^2 = 25/16 \approx 1.56$ 個の画素しか登録されないため，適切なヒストグラムが生成されていない可能性も高い．一方，後者では，平均で一つのビンあたり $S_0^2/R^2 = 25^2/8^2 = 625/64 \approx 9.77$ 個の画素が登録されることから，妥当なヒストグラムが得られていると考えられる．以上のことから，パッチサイズとビンの

個数の関係は結果に大きく影響するものと予想される。これに関しては5.6節の実験で考察する。

5.2 パッチサイズ変化率の影響

パッチサイズ変化率 G の影響を調べた実験の結果を図 6 に示す (5.3 節で図 7 と比較するため、上位選択数 T の値も表記している)。(d), (e), (f) は $G = 0.5$ としたときの結果であり、不適切な相対配置が得られている。一方, (g), (h), (i) は $G = 0.8$ としたときの結果であり、前者に比べて良好な結果が得られている。4.4 節の最後で述べたように、変化率が $G > 1/2$ の場合には、解像度の減少に伴って一つのパッチが覆う入力画像内の範囲は相対的に大きくなっていく。この場合、2枚の入力画像間で類似する領域を探すという目的に対して、はじめは低解像度で相対的に大きな範囲内で探し、徐々に相対的な範囲を狭めていくことで、効果的に類似する領域を探索できていると考えられる。

5.3 上位選択数の影響

上位選択数 T の影響を調べた実験の結果を図 7 に示す。図 6 の(d), (e), (f) に示した $G = 0.5$, $T = 5$ の結果に対して、図 7 は同じ入力画像に対して $G = 0.5$, $T = 15$ とした結果である。前者に比べて後者は良好な結果が得られている。候補とする相対配置の数を増やすことで、良好な結果が得られることがわかった。

5.4 解像度レベル数の影響

解像度レベル数 K の影響を調べた実験の結果を図 8 に示す。(d), (e), (f) は $K = 4$, (g), (h), (i) は $K = 2$ としたときの結果である。出力画像に大きな変化は見られないが、最適な相対配置を決定するまでに要した処理時間は、前者が 41.180 秒、後者が 561.279 秒であり、解像度レベル数の減少が大きく処理時間を増加させた。これは、解像度レベル数が小さな場合には最低解像度が高く、4.3 節のアルゴリズムのステップ 2 における 1 画素刻みの全ての相対配置に対する最類似パスの探索に大きな処理時間を要することが主な原因と考えられる。解像度レベル数を大きくすることにより、短時間で良好な結果が得られることが確認できた。

5.5 明度 L の有無の影響

本手法では、4.2 節で述べたように、パッチ間の類似度の評価に Lab 表色系の a, b 値による 2 次元ヒストグラムを用いている。そこで、明度 L の有無の影響を調べるため、a, b 値に明度 L も含めた 3 次元ヒストグラムを用いた場合の結果を図 9 に示す。同じ解像度レベル数 $K = 4$ において、図 8 の(d), (e), (f) は a, b 値のみ、図 9 は同じ入力画像に対して明度 L も含めた場合の結果である。この実験では、明度 L の有無によって全く異なる結果となっている。それぞれ、一部に不自然さが見られるものの、妥当な出力画像が得られており、優劣をつけることはできない。明度 L の有無がどのように結果に影響するかは入力画像に依存するものと思われる。

表 1. 一つのビンあたりの平均登録画素数 S_0^2/R^2

		R		
		4	8	16
S_0	15	14.06	3.52	0.88
	25	39.06	9.77	2.44
	35	76.56	19.14	4.79

5.6 パッチサイズに対するヒストグラムのビンの分割数の影響

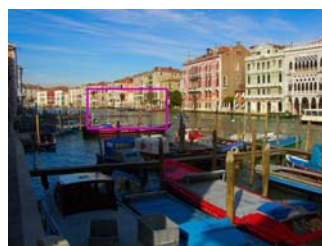
本手法では、2枚の入力画像間で類似する画素色のパターンを持つ領域を探すという目的のため、パッチを単位として色の分布を比較するカラーヒストグラムを用いた類似度の評価を行う。これに関して、5.1 節で述べたパッチサイズとヒストグラムのビンの個数の関係について、図 5 と同じ入力画像に対して実験した結果による出力画像を図 10 に示す。この実験では、初期パッチサイズを $S_0 = 15, 25, 35$ 、ヒストグラムのビンの分割数を $R = 4, 8, 16$ とし、それぞれの組み合わせに対して本手法を適用し、得られた最適な相対配置による出力画像を生成している。最高解像度レベル $k = 0$ における一つのビンあたりの理論上の平均登録画素数 S_0^2/R^2 を表 1 に示す。まず、 $R = 16$ の場合には、 S_0 の値がいずれの場合でも、良好な相対配置が得られていないことが分かる。この原因としては、ビンの分割数 R が大きな場合には、ビンの個数が多すぎ、2枚の入力画像間で類似する色値を持つ画素が互いに近くの異なるビンに登録されてしまい、結果として式(4)によるヒストグラムの類似度が高い値とならないことが起きうることが考えられる。また、今回の実験では、表 1 から、 $R = 16$ の場合には一つのビンあたりの平均登録画素数が少なく、適切なヒストグラムが生成されていない可能性も高い。次に、 $R = 4$ の場合には、 $R = 16$ の場合に比べて妥当な相対配置が得られてはいるが、 S_0 の値がいずれの場合でも出力画像が不自然な部分を持っている。これは、2枚の入力画像を自然につなぐ適切な境界領域が得られていないことを意味し、適切な最類似パスが得られていないことが原因と考えられる。表 1 から、 $R = 4$ の場合の一つのビンあたりの平均登録画素数は妥当であると考えられるが、ビンの分割数 R が小さな場合には、ビンの個数が少なすぎるという問題があり、2枚の入力画像間で全く異なる色値を持つ画素が同じビンに登録されてしまうことが起きう。その場合、結果として、実際には類似度が低いパスが最類似パスとして得られてしまうことになる。最後に、 $R = 8$ の場合であるが、 $S_0 = 15$ の場合には一つのビンあたりの平均登録画素数が少ないものの、 S_0 の値がいずれの場合でも良好な出力画像が得られている。この実験からは、いずれの初期パッチサイズにおいても、ビンの分割数を $R = 8$ とした場合に良好な結果が生じており、よって、本論文の他の実験でも $R = 8$ をデフォルトの値とした。

5.7 画素ベース探索法との比較

本研究では、パッチベースイメージキルティング探索法を提案している。この有効性を検証するため、3 節で述べたイメージキルティング法で用いられている従来の画素ベースの探索法 [1] を用いた場合と比較した実験の結果を図 11 に示す。画素ベ



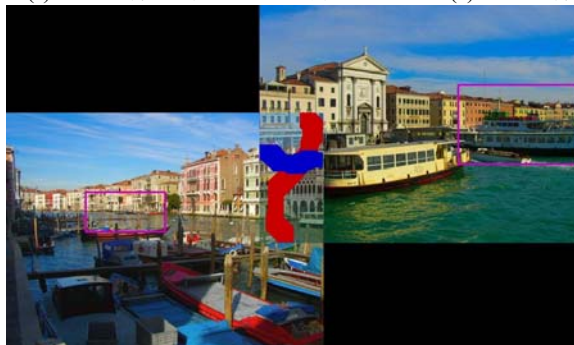
(a) 入力画像 1 (480×360 画素)



(b) 入力画像 2 (480×360 画素)



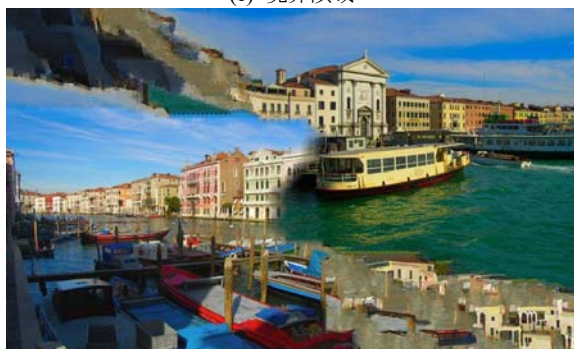
(c) 入力画像 2 (明度 L 調整後)



(d) 最適な相対配置と最類似パス



(e) 境界領域



(f) 出力画像

(d), (e), (f) パッチサイズ変化率 $G = 0.5$, 上位選択数 $T = 5$

図 6. パッチサイズ変化率の影響



(g) 最適な相対配置と最類似パス



(h) 境界領域



(i) 出力画像

(g), (h), (i) パッチサイズ変化率 $G = 0.8$, 上位選択数 $T = 5$



(a) 最適な相対配置と最類似パス



(b) 境界領域



(c) 出力画像

図 7. 上位選択数の影響 パッチサイズ変化率 $G = 0.5$, 上位選択数 $T = 15$

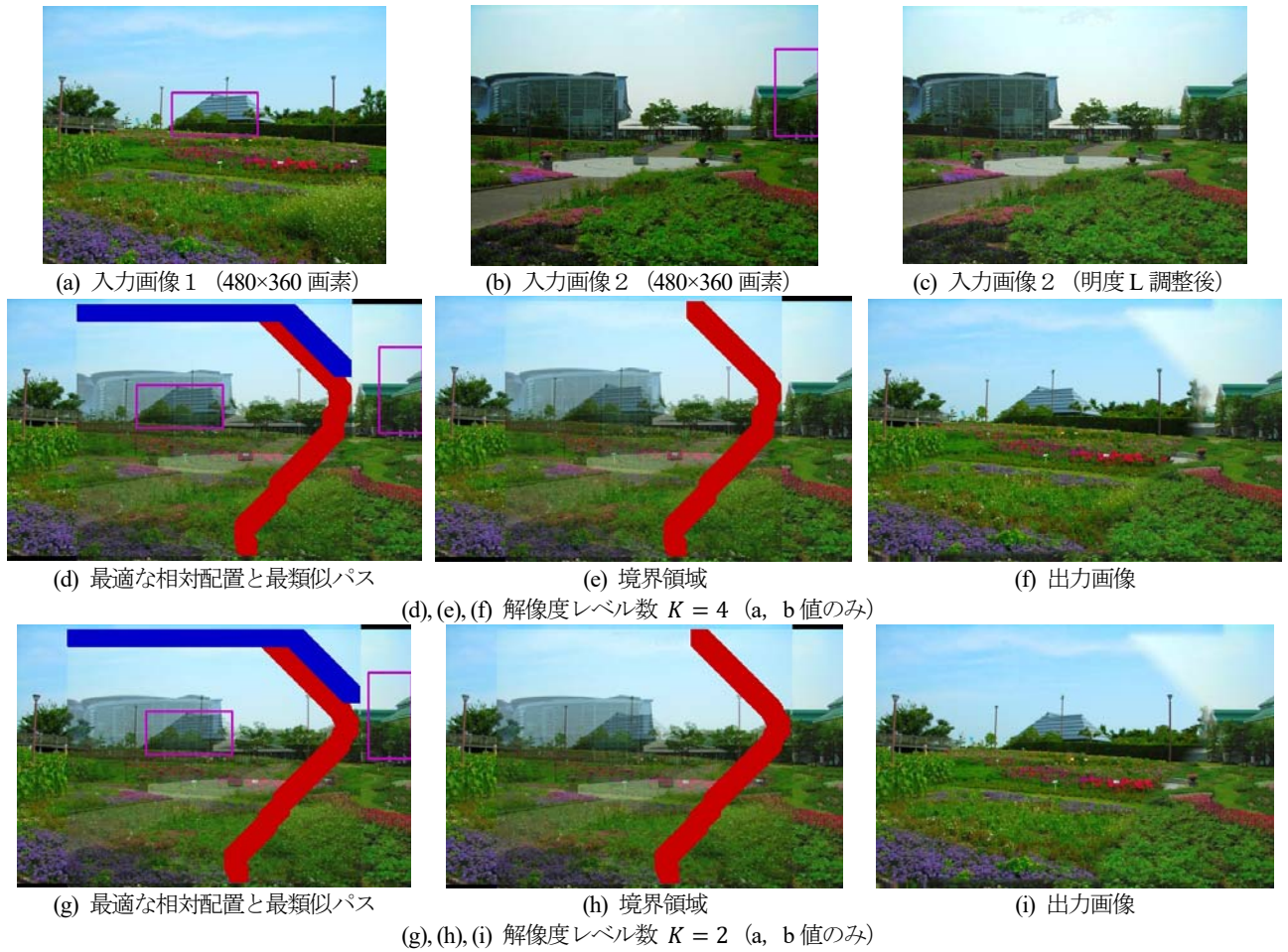


図 8. 解像度レベル数の影響

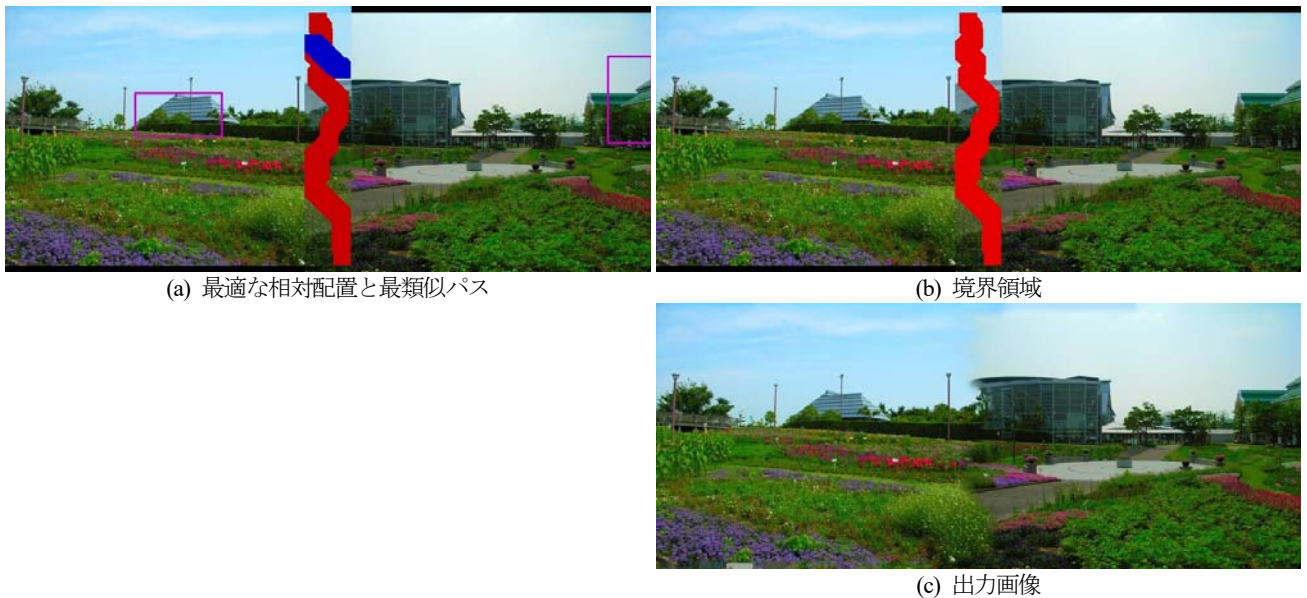


図 9. 明度 L の有無の影響 明度 L を含めた場合 (解像度レベル数 $K = 4$)

ースの探索法については、式(1)に示す画素間の類似性を比較する相違度 (SSD) に用いる色値として、式(1)と同じ RGB 値を用いるだけでなく、パッチベースの探索法と同じ Lab 表色系の a, b 値も用いた。図 8 に示したパッチベース探索法の結果に対し

て、同じ入力画像を用いて、図 11 の(a), (b), (c)は RGB 値による画素ベース探索法、(d), (e), (f)は a, b 値による画素ベース探索法で得られた結果である。図 8 の 2 枚の入力画像と図 11 の (a), (b), (d), (e)を観察すると、いずれの画素ベース探索法の結



(a) $S_0 = 15$, $R = 4$



(b) $S_0 = 15$, $R = 8$



(c) $S_0 = 15$, $R = 16$



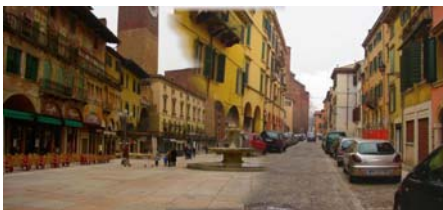
(d) $S_0 = 25$, $R = 4$



(e) $S_0 = 25$, $R = 8$



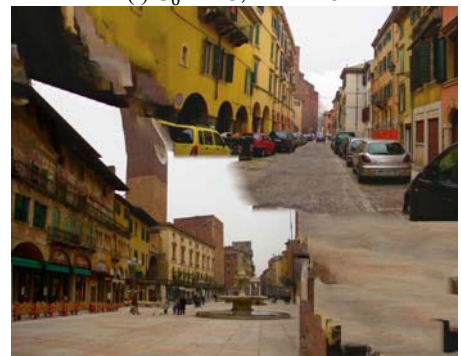
(f) $S_0 = 25$, $R = 16$



(g) $S_0 = 35$, $R = 4$



(h) $S_0 = 35$, $R = 8$



(i) $S_0 = 35$, $R = 16$

図 10. パッチサイズに対するヒストグラムのビンの分割数の影響

果においても、入力画像 1 内に写っている 4 本の柱のうちの一つ右側の柱と入力画像 2 内の左側の大きな建物が重なった部分を最類似パスが通過しており、その部分が境界領域となっている。これは、画素ベース探索法によって、その部分の類似度が高いものと判定されたことを意味している。しかし、明らかに本研究で目的とされる境界領域としては適切なものではなく、結果として得られる図 11 の(c), (f)に示す出力画像では、その建物が途中で不自然に切れてしまっている。すなわち、画素ベース探索法による 1 画素幅の類似性の評価では、類似する領域であるかどうかを十分には評価できていないと言える。一方、図 8 の結果から、縦横方向に数画素幅を持つパッチを評価の単位とし、さらに、カラーヒストグラムによって色の分布の類似性を評価するパッチベース探索法は良好な結果を生み出し、有効に働いていることが確認できる。

5.8 パッチ間の SSD による類似度評価との比較

本手法では、色の分布が似ていることを類似度の指標とする

ため、2つのパッチ間の類似度の評価に Lab 表色系の a, b 値によるヒストグラムを用いた。この有効性を検証するため、式(1)に示す SSD, すなわち、2つのパッチ間で同じ位置で重なる画素どうしの色値の差の二乗をパッチ内で総和した値を相違度として用いて類似性を評価する場合と比較を行った。5.7 節と同様、式(1)と同じ RGB 値を用いるだけでなく、ヒストグラムの場合と同じ Lab 表色系の a, b 値も用いた。実験の結果を図 12 に示す。(d), (e), (f)はヒストグラム, (g), (h), (i)は RGB 値による SSD, (j), (k), (l)は a, b 値による SSD を用いた場合の結果である。ヒストグラムを用いた場合には、入力画像 2 の右側の草原の部分につながるように入力画像 1 の草原が合成されており、自然な出力画像が生成されている。一方、SSD を用いた場合には、RGB 値と a, b 値のいずれでも、入力画像 2 の道路の左側に入力画像 1 の草原が合成されてしまっており、生成された出力画像は不自然な印象がある。SSD を用いた類似度の評価では、2つのパッチ間で同じ位置で重なり合う画素どうしが類似色を持つことを基準とするが、異なるシーンの入力画像で

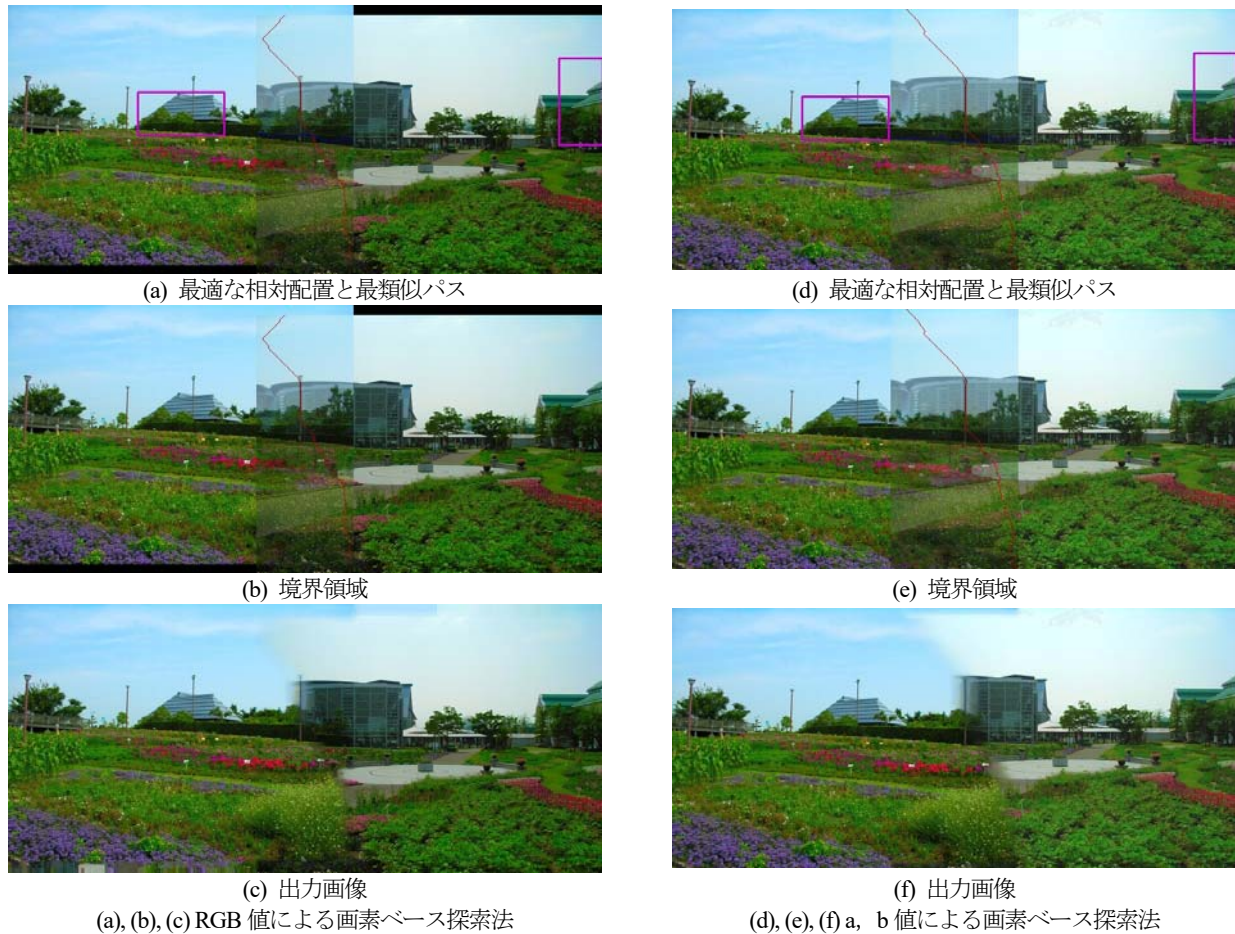


図 11. パッチベース探索法と画素ベース探索法の比較

は、2つのパッチ間、さらには、複数のパッチが連続するパスの全体に渡って重なり合う画素どうしが類似色を持つことは稀有であるため、適切な評価がなされていないと考えられる。一方、ヒストグラムを用いた類似度の評価は、色の分布が似ていることを基準としており、異なるシーンの入力画像間で類似する部分を探すという目的に合っているとと言える。

5.9 保持領域の影響

図 13 は、出力画像上にそのまま複製するために入力画像上で指定する保持領域の影響を調べた実験結果である。同じ入力画像に対して、(d), (e), (f)は保持領域がある場合、(g), (h), (i)はない場合である。それぞれの場合で異なる最適な相対配置と出力画像が生成されている。4.5 節で述べたように、保持領域は最適な相対配置を決定する際の制約となり、保持領域を通過しないように最類似パスが決定される。この実験結果の後者では、前者で保持領域として指定した入力画像 1 上の大きな岩を最類似パスが通過した結果、大きな岩の一部が保持されず、ほぼ半分が欠落した出力画像となっている。保持したい領域がある場合、保持領域の指定が有効であることが分かる。

5.10 出力画像の合成方法の比較

図 14 は、出力画像を完成させるための境界領域に対する合成処理として、4.7 節で述べた 2つの方法を適用した結果であ

る。(a1), ..., (e1)は境界領域にフェザリングとテクスチャ合成を適用して得られた出力画像であり、それぞれ、各画像の下に表記したように、これまでの他の図中で既出のものである。一方、(a2), ..., (e2)は、それぞれ、同じ境界領域に対してボアソン画像合成を適用して得られた出力画像である。以下、関連する既出の図中の「境界領域」の画像を参照することで、境界領域の部分に注目して議論を進める。

フェザリングとテクスチャ合成による方法は、4.7 節で述べたように、まず、フェザリングによって境界領域内で重なる 2枚の入力画像を境界からの距離に応じて線形にブレンドし、その後、ブレンドによる色パターンをテクスチャ合成によって入力画像上の実在するパターンで置き換えることで、境界領域をより自然に見せることを目標としている。しかし、図 14 の(a1), ..., (e1)は、それぞれに妥当な結果が得られてはいるものの、不自然にぼやけた部分も多く見られる。反復法によるテクスチャ合成によって境界領域内の画素色の置き換えを繰り返しても、この状況は改善されなかった。この原因は以下のように考えられる。境界領域では 2枚の入力画像内の互いに類似する部分どうしが重なり合っているが、それらの入力画像は異なるシーンを持つため、実際に映っているのは異なるものである。また、ヒストグラムの類似度で選ばれるのは色の分布が類似している部分であり、重なり合う画素どうしが類似色を持つわけではない。その結果として、多くの場合、フェザリングによるブレン

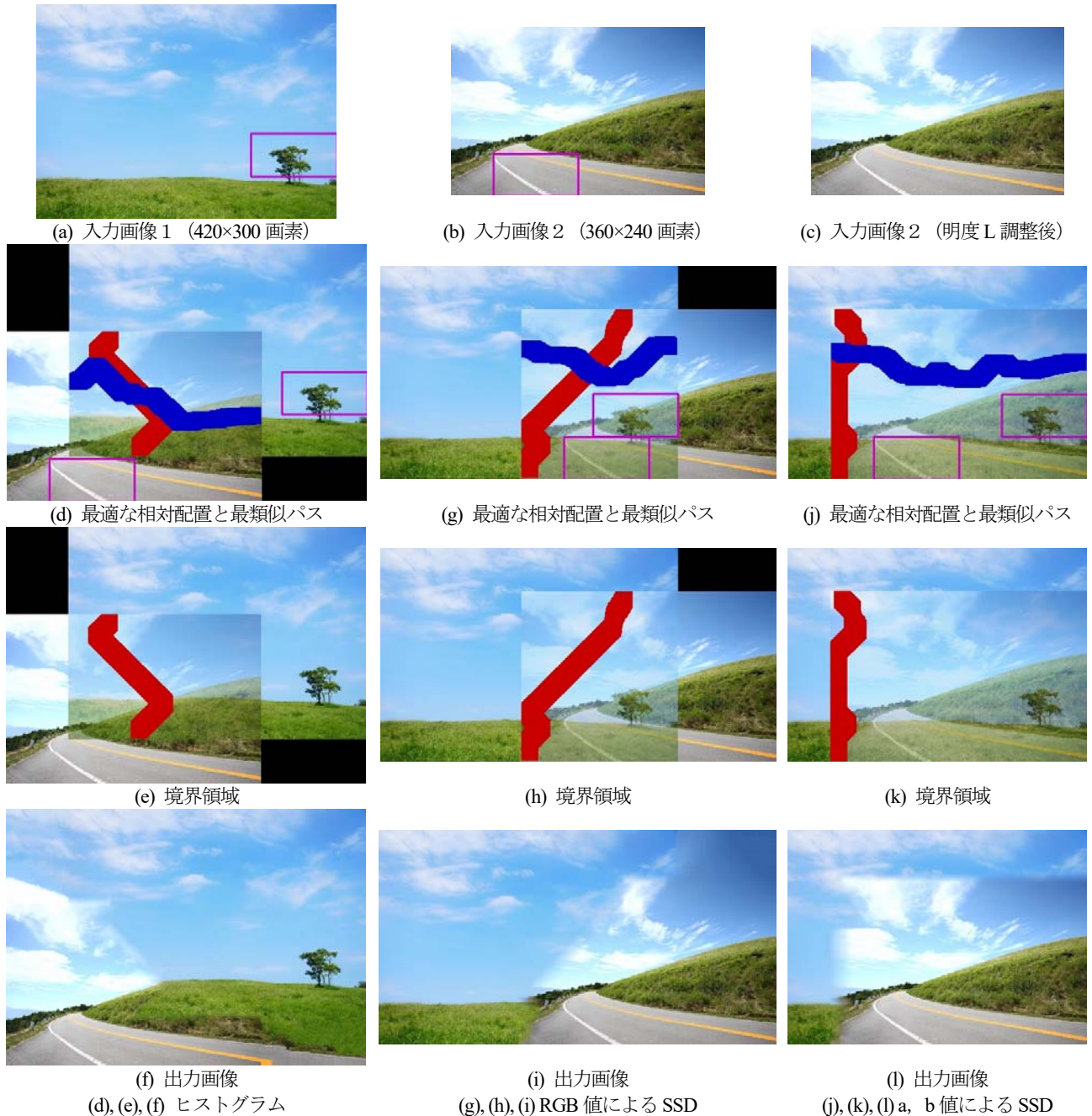


図 12. ヒストグラムと SSD によるパッチ間の類似度評価の比較

ドで生じる色パターンはいずれの入力画像上の元のパターンからも離れたものになってしまう。そのため、反復法によるテクスチャ合成において、ブレンドによる色パターンに類似する適切な近隣画素群、特に、物体の輪郭を鮮明に表すような近隣画素群を入力画像上で探索することが難しく、入力画像上に実在する鮮明なパターンへの適切な置き換えができないものと考えられる。詳細な観察のため、図 6 の(g), (h), (i)に関して、図 15 の(a)は境界領域に対してフェザリングのみを行った場合、(c)はそれを初期状態として反復法によるテクスチャ合成を行った場合（図 6 (i)と同じ画像）を示す。図 6 の 2 つの入力画像のそれぞれに写る白い建物が境界領域内で重なり合い、合成されているが、それぞれ、その屋根の部分（図中の黄色枠）を拡大した

ものが(b)と(d)である。フェザリングによってブレンドされた色パターンがテクスチャ合成によってぼかされていることがわかる。入力画像上に実在する適切なパターンに置き換えるための有効な方法の検討が今後の課題である。また、処理時間は、フェザリングのみの場合が 0.105 秒であるのに対して、テクスチャ合成までを行った場合が 3 分 36.235 秒であった。テクスチャ合成により大幅に処理時間が増加した原因は、反復ごとに各入力画像内の全ての近隣画素群（この実験では 7×7 画素）を評価して最適なものを探索しているためである。最適な近隣画素群を効率的に探索する方法の検討も今後の課題である。

一方、ポアソン画像合成による図 14 の(a2), ..., (e2)では、上記の境界領域内のぼやけの多くが軽減されている。例えば、(a2)

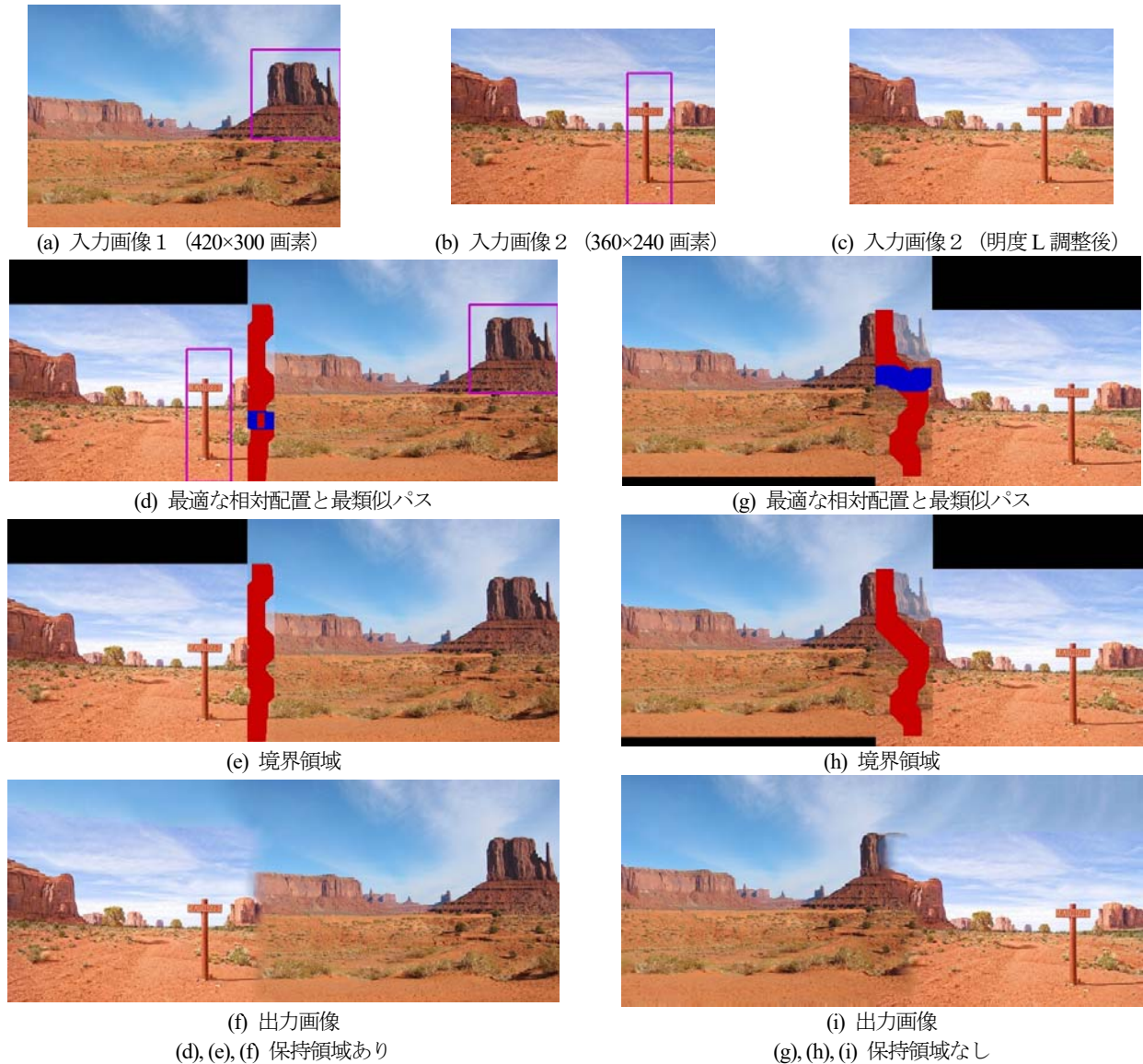


図 13. 保持領域の影響

では、タイル状の路面と細かな石畳の境目が自然に融合されている。(b2)と(c2)では、境界領域に沿った草花のぼやけが軽減されて明瞭になり、また、境界領域内で空の下に位置する木が鮮明になっている。(d2)では2枚の入力画像による左右からの雲と草原の境目、(e2)では地面の境目が自然に融合されている。これらは、2枚の入力画像のパターンのうちでより特徴的なほうを出力画像上に出現させるポアソン画像合成の能力が有効に働いた結果である。しかし、不自然な融合が行われた部分もあり、例えば、(a2)では境界領域に沿った建物の部分が不自然である。同様に、図 15 の(e)は(a)、(c)と同じ境界領域に対してポアソン画像合成を行った結果であり、(f)は(b)、(d)と同じ部分の拡大図であるが、重なり合う2つの白い建物が不自然に合成されている。このような状況は、2枚の入力画像が境界領域内で(画素色の微分の絶対値の大きさという観点で)同程度の特徴的なパターンを持つ場合に起こりうる。また、図 14 の(e2)では、境界領域上で中央に位置する左右の岩山の部分に不自然な印象があり、特に、右側の岩山は左側の空と不自然に融合している。こ

れは求められた最適な相対配置そのものに原因がある。異なるシーンの入力画像間では、境界領域全体に渡って高い類似度を持つ相対配置を得ることは難しいため、類似度が低い部分に対してもなるべく不自然さが目立たないような合成を行う方法の検討が必要である。

5.11 手動による微調整の導入

本研究では、自動的に2枚の入力画像の最適な相対配置を決定する手法を提案している。しかし、実際の利用においては、満足のいかない結果が得られた場合には、手動で微調整することが有用である。はじめから手動で類似する領域が重なる相対配置を求めることに比べて、本手法で自動的に得られた相対配置から手動で微調整することにより、少ない負担で良好な相対配置を得ることができる。このような手動による微調整を行った実験の結果を図 16 に示す。図 16 の(a)は本手法で自動的に得られた図 7 の(c)と同じ画像であり、図 16 の(b)は同じ境界領域にポアソン画像合成を適用した結果である。それに対して、図



(a1) フェザリング, テクスチャ合成 (図 5 (i))



(a2) ポアソン画像合成



(b1) フェザリング, テクスチャ合成 (図 8 (f))



(c1) フェザリング, テクスチャ合成 (図 9 (c))



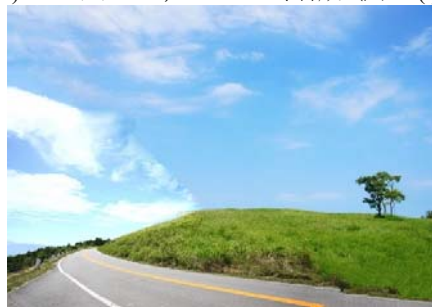
(b2) ポアソン画像合成



(c2) ポアソン画像合成



(d1) フェザリング, テクスチャ合成 (図 12 (f))



(d2) ポアソン画像合成

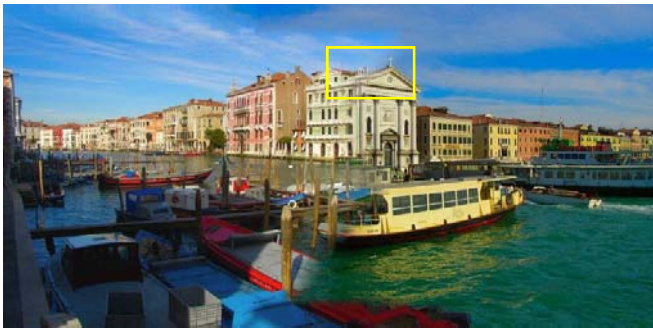


(e1) フェザリング, テクスチャ合成 (図 13 (f))



(e2) ポアソン画像合成

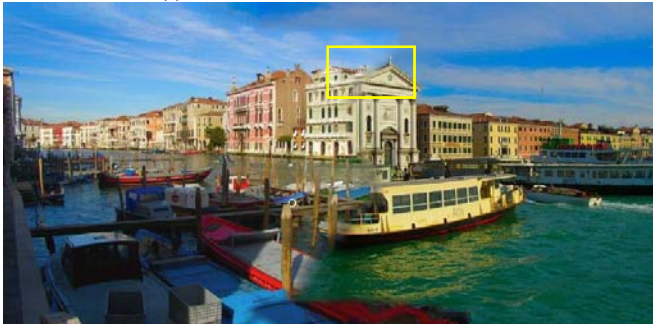
図 14. 出力画像の合成方法の比較



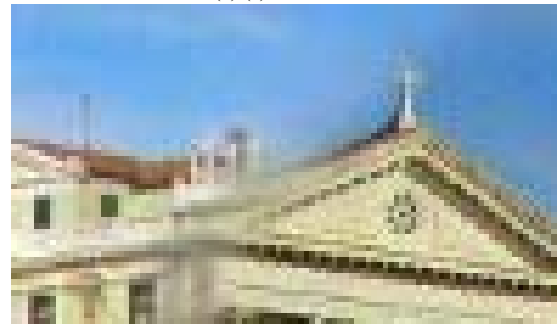
(a) フェザリングのみを行った場合



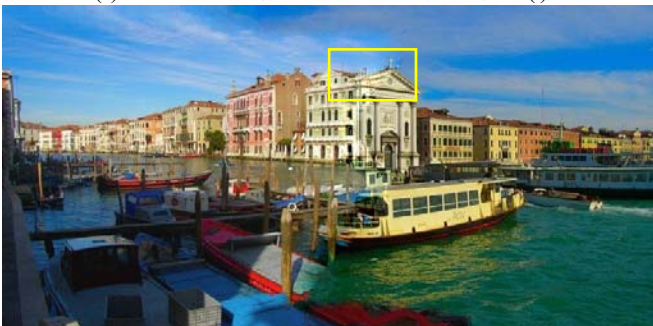
(b) (a)の拡大図



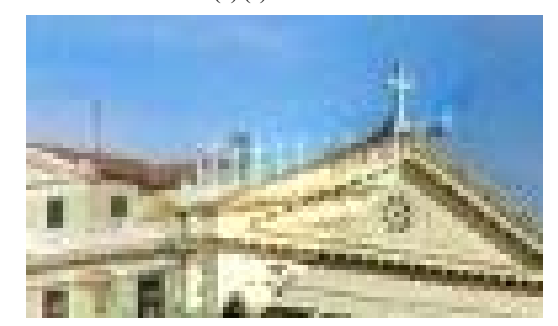
(c) テクスチャ合成までを行った場合 (図6(i))



(d) (c)の拡大図



(e) ポアソン画像合成を行った場合



(f) (e)の拡大図

図 15. フェザリング, テクスチャ合成, ポアソン画像合成の比較

16の(c), (d), (e), (f)は手動による微調整を加えた結果であり, 自動的に得られた最適な相対配置から横方向に0画素, 縦方向に-15画素だけずらしている. 出力画像上の中央部の白い建物の屋根の部分(図中の黄色丸)に注目すると, 手動調整によって, より良好な結果を得ることができている. しかし, 出力画像の下部にある赤い舟が不自然に途切れてしまっている. この解決法としては, 途切れて欲しくない部分を保持領域として指定することが考えられる. しかし, 出力画像上に残す必要のない部分まで保持領域として指定することは, 特に, 途切れて欲しくない多くの物体を含むような入力画像では煩雑な作業である. そこで, 境界領域の手動による微調整を可能とすることも一案であり, その実現方法は今後の課題の一つである.

6. 結論

本研究では, カラーヒストグラムを利用してパッチ単位で類似度を評価することで最類似パスを探索するパッチベースイメージキルティング探索法を用いて, 2枚の入力画像間の最適な相対配置を決定して自然な出力画像を合成するイメージモンタ

ージュ法を提案した. そして, 条件を変えた様々な実験により, 異なるシーンの入力画像から自然な出力画像が合成されることを確認し, 本手法の有効性を検証した.

今後の課題としては, 実験結果に関連して5節でも述べたように, 境界領域を自然に合成するためのテクスチャ合成とポアソン画像合成の改善, テクスチャ合成の高速化, 手動による相対配置や境界領域の微調整機能が挙げられる. また, 入力画像の拡大・縮小の導入, 不要部分の自動削除なども挙げられる. さらに, 3枚以上の入力画像への拡張も予定している. また, 出力画像の定量的な評価方法の検討も課題である.

参考文献

- [1] A. A. Efros and W. T. Freeman, Image Quilting for Texture Synthesis and Transfer, Proc. of ACM SIGGRAPH 2001, pp. 341-346, 2001.
- [2] R. Szeliski and H.-Y. Shum, Creating Full View Panoramic Image Mosaics and Environment Maps, Proc. of ACM SIGGRAPH 1997, pp. 251-258, 1997.



(a) フェザリングとテクスチャ合成による出力画像 (図 7(c))



(b) ポアソン画像合成による出力画像

(a), (b) 自動的に得られた結果



(c) 手動による微調整を加えた相対配置と最類似パス



(d) 境界領域



(e) フェザリングとテクスチャ合成による出力画像



(f) ポアソン画像合成による出力画像

(c), (d), (e), (f) 手動による微調整を加えた結果

図 16. 手動による微調整の導入

- [3] V. Kwatra, I. Essa, A. Bobick and N. Kwatra, Texture Optimization for Example-based Synthesis, Proc. of ACM SIGGRAPH 2005, pp. 795-802, 2005.
- [4] P. Perez, M. Gangnet and A. Blake, Poisson Image Editing, Proc. of ACM SIGGRAPH 2003, pp. 313-318, 2003.
- [5] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin and M. Cohen, Interactive Digital Photomontage, Proc. of ACM SIGGRAPH 2004, pp. 294-302, 2004.
- [6] M. Johnson, G. J. Brostow, J. Shotton, O. Arandjelovic, V. Kwatra and R. Cipolla, Semantic Photo Synthesis, Proc. of EUROGRAPHICS 2006, pp. 407-413, 2006.
- [7] V. Kwatra, A. Schodl, I. Essa, G. Turk and A. Bobick, Graphcut Textures: Image and Video Synthesis Using Graph Cuts, Proc. of ACM SIGGRAPH 2003, pp. 277-286, 2003.
- [8] J. Jia, J. Sun, C. Tang and H. Shum, Drag-and-Drop Pasting, Proc. of ACM SIGGRAPH 2006, pp. 631-637, 2006.
- [9] N. Diakopoulos, I. Essa and R. Jain, Content Based Image Synthesis, Conf. on Image and Video Retrieval 2004, pp. 299-307, 2004.
- [10] J. Hays and A. A. Efros, Scene Completion Using Millions of Photographs, Proc. of ACM SIGGRAPH 2007, pp. 4(1)-(7), 2007.
- [11] H. Kang, Y. Matsushita, X. Tang and X. Chen, Space-Time Video Montage, Proc. of IEEE Computer Vision and Pattern Recognition 2006, vol. 2, pp. 1331-1338, 2006.
- [12] D. Simakov, Y. Caspi, E. Shechtman and M. Irani, Summarizing Visual Data Using Bidirectional Similarity, Proc. of IEEE Computer Vision and Pattern Recognition 2008, pp. 1-8, 2008.
- [13] S. Avidan and A. Shamir, Seam Carving for Content-Aware Image Resizing, Proc. of ACM SIGGRAPH 2007, pp. 10(1)-(9), 2007.
- [14] J. Sivic, B. Kaneva, A. Torralba, S. Avidan and W. T. Freeman, Creating and Exploring a Large Photorealistic Virtual Space, Proc. of IEEE Computer Vision and Pattern Recognition Workshops 2008, pp. 1-8, 2008.

- [15] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman and P. Sen, Image Melding: Combining Inconsistent Images using Patch-based Synthesis, Proc. of ACM SIGGRAPH 2012, pp. 82(1)-(10), 2012.
- [16] C. Rother, S. Kumar, V. Kolmogorov and A. Blake, Digital Tapestry, Proc. of IEEE Computer Vision and Pattern Recognition 2005, vol. 1, pp. 589-596, 2005.
- [17] C. Rother, L. Bordeaux, Y. Hamadi and A. Blake, AutoCollage, Proc. of ACM SIGGRAPH 2006, pp. 847-852, 2006.
- [18] M. J. Swain and D. H. Ballard, Color Indexing, International Journal of Computer Vision, vol. 7, no. 1, pp. 11-32, 1991.
- [19] F. C. Crow, Summed-area Tables for Texture Mapping, Proc. of ACM SIGGRAPH 1984, pp. 207-212, 1984.
- [20] P. Viola and M. Jones, Rapid Object Detection using a Boosted Cascade of Simple Features, Proc. of IEEE Computer Vision and Pattern Recognition 2001, pp. I-511-518, 2001.
- [21] <http://www.photo-ac.com/>
- [22] <https://pixabay.com/>
- [23] <http://www.futta.net/>
- [24] <http://www.beiz.jp/>
- [25] <https://pro.foto.ne.jp/>
- [26] 鈴木康太, 藤本忠博, パッチベースイメーজキルティング探索法を用いた2画像の最適な相対配置によるイメージモンタージュ法, NICOGRAPH 2016, C-8, pp. 1-8, 2016.

鈴木 康太



2015年岩手大学工学部卒業。2017年同大学大学院工学研究科博士前期課程修了。在学中は知的映像編集に関する研究に従事。現在、三菱電機インフォメーションネットワーク株式会社勤務。

藤本 忠博



1990年慶應義塾大学理工学部卒業。1992年同大学大学院理工学研究科前期博士課程修了。同年(株)三菱総合研究所入社。1995年同研究所退職。同年慶應義塾大学大学院理工学研究科後期博士課程入学。1999年同大学院単位取得退学。同年岩手大学助手。2000年博士(工学)(慶應義塾大学)。2002年岩手大学講師。2005年助教授。2007年准教授。2016年教授。コンピュータグラフィックス, コンピュータビジョン, 画像処理に関する研究に従事。ACM, IEEE, 芸術科学会, 他会員。