

AR マーカ除去のための実時間背景画像変形

河合 紀彦^{*1}

佐藤 智和^{*1}

横矢 直和^{*1}

Abstract – 本稿では、拡張現実感で用いられる AR マーカの違和感のない視覚的除去を実現するための、背景画像の実時間変形手法を提案する。従来、AR マーカの実時間除去は、マーカが平面上に存在するという仮定のもと、カメラ位置姿勢に基づく単一の射影変換行列を用いて、背景画像を変形し、マーカ上に重畳することで実現していた。しかし、マーカが完全な平面上に配置されていない場合や、マーカが印刷された物体の厚みのためにマーカが存在する平面と周辺の平面に位置ずれがある場合には、背景画像を射影変換で変形し合成した際、マーカ領域とその周辺の領域でテクスチャのずれが生じ、違和感が生じる問題がある。そこで本研究では、マーカ周辺のテクスチャの動きに基づきマーカ領域のテクスチャの動きを推定することで、領域境界でのテクスチャのずれを軽減し、違和感のないマーカの除去を実現する。

Keywords : マーカ除去, 隠消現実感, テクスチャ変形

1 はじめに

カメラ付きの PC, スマートフォン, タブレット端末, ポータブルゲーム機の普及により, 現実環境を撮影した映像に仮想物体を重畳することで, 様々な情報を付加する拡張現実感 (Augmented Reality; AR) を用いたアプリケーションが普及している。このようなアプリケーションの一部では, 導入の容易さ, カメラ位置姿勢推定の頑健さ, 仮想物体の配置の容易さ等から, 人工的な正方形マーカを用いることで, 実環境と仮想物体との映像中での融合を実現している [1]。しかし, マーカを用いた拡張現実感では, マーカが常に映像中に写りこむ必要があり, マーカの存在により, マーカ以外の実環境と重畳される仮想物体との間でシームレスな融合が実現できず, ユーザに違和感を与える。この問題に対し, マーカを実時間で映像から除去する手法が提案されている。

Siltanen ら [2] は, フレーム毎にマーカ周辺の特定位置の平均画素値を用いることで背景画像を生成している。この手法では, 毎フレーム高速に背景画像を生成できる一方, 単純な背景画像の生成手法のため, 違和感のあるテクスチャが生成される場合が多い。これに対して, 我々 [3] は, マーカは平面上に配置されているという仮定のもと, キーフレームに対して画像修復 (Image Inpainting) 手法を適用して背景画像を生成し, それを毎フレーム, 射影変換による幾何変形および周辺の輝度値の変化に基づく背景画像の光学的補正を行うことで, マーカ上に違和感のない背景画像を合成し, 視覚的除去を実現している。しかし, 背景形状が平面でない場合には, マーカ領域とその周辺との間でテクスチャのずれが生じる問題がある。

一方, 除去対象をマーカに限らず, 映像中から実時間で不要な実物体を視覚的に取り除く隠消現実感 (Diminished Reality; DR) 手法もマーカの除去に応用することができる。マーカの除去に適用可能な隠消現実感手法として, 事前に実際の背景を撮影する手法, 画像修復を用いる手法がある。

事前に実際の背景を撮影する手法 [4, 5, 6] では, 予めシステムの提供者・ユーザが隠消現実感を利用したい場所において事前撮影する, またはインターネット等から同一シーンを撮影した写真を取得することで, 背景が撮影された参照画像を収集し, これらをユーザの映像中で不要物体上に重畳することで除去を実現している。Cosco ら [4] は, 背景形状が既知という前提のもと, 事前撮影画像をその形状を用いて背景画像のレンダリングを行っている。森ら [5] らは, 対象が 2 つ以下の平面で構成されているという仮定のもと, 特徴点のマッチングにより各平面の射影変換行列を算出し, 背景画像を生成している。Li ら [6] は, 現在のフレームと撮影位置に近い画像を探索し, 単一の射影変換行列を用いて変形し, 背景画像を生成している。これらの手法では, 背景形状が既知であるか高精度に推定されている, または, 空間的に密に多くの箇所から対象シーンを撮影した画像群が存在する必要がある。そうでない場合には, 対象領域とその周辺でテクスチャのずれが生じる。

画像修復を用いる手法は, 毎フレームテクスチャを生成する手法 [7, 8] と, キーフレームにおいて背景画像を生成し, それをフレーム毎に幾何学的・光学的に更新する手法 [9] に分類できる。前者は, 毎フレーム類似したパターンを対象領域以外から探索し対象領域のテクスチャを更新している。このような手法は, フレーム毎には違和感のない背景画像が生成されている

^{*1}奈良先端科学技術大学院大学情報科学研究科

場合も多いが、フレーム間でテクスチャの幾何模様が徐々に変化していくような、時間的なテクスチャの不整合がみられるという問題がある。一方、フレーム毎に幾何学的・光学的に背景画像を更新する手法では、背景が単一、または複数の平面で構成されているという仮定のもと、カメラ位置姿勢推定に基づく射影変換行列により背景画像を変形したうえで、光学的な補正を行い重畳を行っている。しかし、対象の背景形状が仮定を満たさない場合には、対象領域とその周辺でテクスチャのずれが生じる問題がある。

本研究では、AR アプリケーションにおいて任意の場所で気軽に環境中に固定されたマーカを視覚的に除去する状況を想定し、配置する前に1枚の静止画像を撮影する、または画像修復を適用することにより背景画像を生成し、その背景画像をフレーム毎に幾何学的・光学的に補正し、対象領域に重畳するアプローチを採る。その中でも、本研究では、マーカが完全な平面上に配置されていない場合や、マーカが印刷された物体の厚みのためにマーカが存在する平面と周辺の平面に位置ずれがある場合に生じる対象領域とその周辺でテクスチャのずれを、対象領域周辺のテクスチャの動きに基づき対象領域内部の背景画像の幾何模様を補正することで、テクスチャのずれによる違和感を低減したマーカ除去手法を提案する。

2 AR マーカ除去の概要

図1に、本研究の提案するマーカ除去の流れを示す。提案手法では、まず事前撮影または画像修復によって、マーカを配置する箇所の背景画像を取得する。図2に示すように、事前撮影の場合には、まず1枚画像を撮影し(a-1)、カメラを動かさないうままマーカを配置する(a-2)。次に、マーカを真正面から見たような正対画像の見えになるよう撮影した背景画像を射影変換し正対背景画像Bを生成する(a-3)。画像修復の場合には、まずマーカを配置して画像を取得し(a'-1)、それをマーカを真正面から見た画像に変換する(a'-2)。次に、画像修復を適用することで(a'-3)、正対背景画像Bを生成する[3]。その後、毎フレーム行う背景画像変形のための前処理として、正対背景画像B中のマーカ周辺領域において特徴点の抽出とその信頼度を算出する(a-4)。なお、画像修復の場合には、文献[3]のように次に述べる毎フレーム処理と並列に行うこともできる。

毎フレームのマーカ除去画像生成・提示処理として、まず画像を取得した後(b-1)、マーカからカメラの位置姿勢を算出し(b-2)、それを用いてマーカを真正面から見たような画像に射影変換する(b-3)。次に、正対化した入力画像と正対背景画像Bのマーカ周辺領

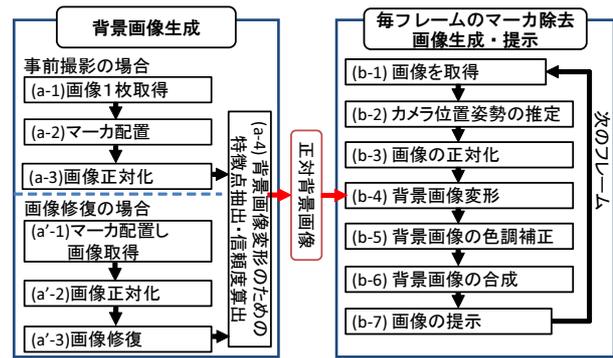


図1 提案手法の処理の流れ

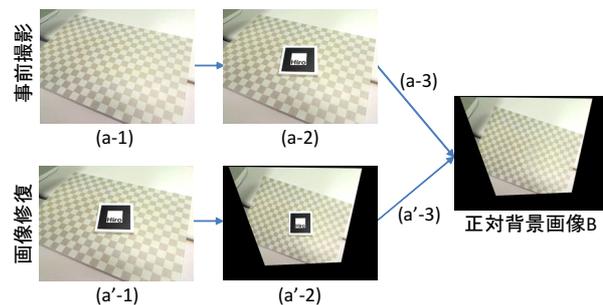


図2 正対背景画像生成の流れ

域で特徴点のマッチングを行い、それに基づき正対背景画像の変形を行う(b-4)。次に、入力画像と背景画像間の明るさや色の差異を補償するため、変形した背景画像を正対入力画像のマーカ領域に合成することを想定したポアソン画像合成[10]により色調補正を行う(b-5)。最後に、もとの入力画像の見えに変換したうえでマーカ上に合成し(b-6)、提示を行う(b-7)。

次節では、正対背景画像B中のマーカ周辺領域 $\partial\Omega$ における特徴点の抽出とその信頼度の算出手法(a-4)、および背景画像変形(b-4)について詳述する。

3 背景画像変形

背景画像変形処理では、正対画像上でのマーカ領域周辺のテクスチャの動きを算出し、それに基づきマーカ領域に対応する正対背景画像Bの動きを推定し、正対背景画像を変形することで、マーカ領域とその周辺領域間のテクスチャのずれを軽減する。以下では、まず背景画像変形のための事前処理として背景画像生成後に行う、背景画像からの特徴点の抽出およびその信頼度の算出について述べる。次に、毎フレームの背景画像変形について述べる。

3.1 背景画像の特徴点抽出と信頼度算出

処理(a-4)では、マーカ領域周辺 $\partial\Omega$ の特徴点の検出およびその信頼度の算出を行う。毎フレーム行う背景画像変形処理では、ここで求める特徴点を用いて、

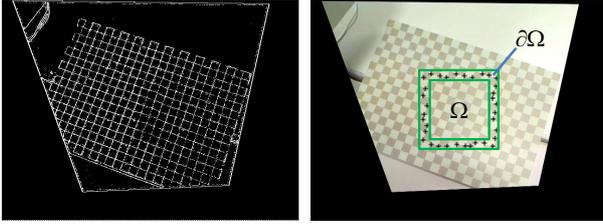


図3 特徴点抽出

背景画像と現フレーム間のマーカ領域周辺の画像の動きを求め、またそれをマーカ領域内の動きの補間に用いるため、対応付けに用いた際の信頼度が高く、また高速処理のために多すぎずかつ補間に用いるのに少なすぎない数の特徴点を検出する必要がある。

具体的な処理として、まずマーカ領域 Ω の境界から距離が l 画素以内の領域を特徴点の検出領域 $\partial\Omega$ として設定する。次に、図3左に示すように、LoG(Laplacian of Gaussian) フィルタを適用し、その零交差点の画素を特徴点候補として抽出する。次に、検出領域 $\partial\Omega$ 内の特徴点候補の全画素に対し、信頼度を算出する。本研究では、 i 番目の画素 x_i の信頼度 $C(x_i)$ は x_i 中心とする一定範囲内に対する局所パッチの自己相関に基づき以下の式により算出する。

$$C(x_i) = \frac{|N_{x_i}| - \sum_c \sum_{x' \in N_{x_i}} NCC_c^{(B,B)}(x_i, x')}{|N_{x_i}|} \quad (1)$$

ただし、 N_{x_i} は x_i を中心とする一定範囲内かつ x_i を除く画素の集合、 $|N_{x_i}|$ は N_{x_i} の要素数である。 $NCC_c^{(B,B)}(x_i, x')$ は、背景画像 B 中の画素 x_i と x' を中心とする局所パッチ同士のチャンネル c (\in RGB) の画素値の正規化相互相関を表し、以下のように定義される。

$$NCC_c^{(B,B)}(x_i, x') = \frac{\sum_{p \in W} I_c^B(x_i + p) I_c^B(x' + p)}{\sqrt{\sum_{p \in W} I_c^B(x_i + p)^2 \sum_{p \in W} I_c^B(x' + p)^2}} \quad (2)$$

ただし、 $I_c^B(x_i)$ は背景画像 B の画素 x_i におけるチャンネル c の画素値、 p はパッチ W 内の画素への遷移ベクトルである。周辺領域に対して自己相関が高い方が、特徴点の信頼度が低くなる。

次に、図3右に示すように、求めた信頼度を用いて、特徴点候補から、信頼度が高くかつ局所的に集中しないよう最終的な特徴点を選出する。ここでは、まず全ての特徴点候補を信頼度の高い順に並び替える。次に、信頼度の高い順に逐次的に以下の条件を満たすかどうかを検定し、満たす場合には特徴点として選出する。

条件1 候補特徴点と既選出の特徴点との距離が L_1 以上 (L_1 : 定数)

条件2 候補特徴点の信頼度 C が距離が L_2 以内の既選出の特徴点の信頼度の $N_2\%$ 以上 (N_2 : 定数)

条件3 候補特徴点の信頼度 C が一番高い信頼度の $N_3\%$ 以上 (N_3 : 定数)

3.2 動きの補間による背景画像変形

背景画像変形処理 (b-4) では、正対背景画像 B 中のマーカ周辺領域 $\partial\Omega$ 内の選出された特徴点において正対背景画像と正対現フレーム画像との対応付け (i) を行い、その変位量の結果に基づきマーカ領域 Ω 内のテクスチャの動きを補間 (ii) することで、背景画像の変形を行う。以下では、 f 番目のフレームにおける背景画像変形について述べる。

(i) マーカ周辺領域における特徴点の対応付け

まず領域 $\partial\Omega$ 内の各特徴点 x_i の対応付けにおいて、空間的に近い各特徴点同士のオプティカルフローの変位量が大きく異ならないようにするため、信頼度の高い特徴点から逐次的に、以下のコストが最小となるよう、 $f-1$ 番目のフレームにおいて対応ついた画素 $h_{f-1}(x_i)$ を中心とする一定範囲 $G_{h_{f-1}(x_i)}$ 内の画素を探索することで、現フレーム f での対応画素 $h_f(x_i)$ を決定する。

$$h_f(x_i) = \arg \max_{x' \in G_{h_{f-1}(x_i)}} \frac{NCC_c^{(B,f)}(x_i, x')}{1 + \sum_{x_j \in M_{x_i}} D_{\|s\|}(\|t\|)} \quad (3)$$

ただし、 $s = x_i - x_j$ 、 $t = x' - h_f(x_j)$ である。ここで、 $NCC_c^{(B,f)}(x_i, x')$ は式 (2) で算出される背景画像 B とフレーム f 間の画素 x_i と画素 x' を中心とするパッチ同士の正規化相互相関である。 M_{x_i} は、画素 x_i を中心とする一定範囲内に存在する特徴点のうち、画素 x_i より信頼度が高く、先に式 (3) により対応づけられた画素の集合である。 $D_{\|s\|}(\|t\|)$ は、特徴点間の距離とオプティカルフローの変位量の差に基づくコスト項であり、以下のように定義する。

$$D_{\|s\|}(\|t\|) = \begin{cases} 0 & (d(\|s\|) > \|t\|) \\ \kappa & (\text{otherwise}) \end{cases} \quad (4)$$

ただし、 $d(\|s\|)$ は $\|s\|$ が大きくなるほど大きい値を取る関数である。後述する実験では、 $d(\|s\|) = 10\|s\|$ とした。 $D_{\|s\|}(\|t\|)$ は、近傍の特徴点同士が同様の動きをすることを促すコスト関数であるが、特徴点同士の距離が遠いほど、オプティカルフローの変位量も大きく異なる傾向があるため、各特徴点間の画素位置の距離 $\|s\|$ が大きくなるほど変位量の差 $\|t\|$ が大きくなることを許容するコスト項になっている。

(ii) 特徴点以外の画素の動き補間

処理 (i) で求めた、各特徴点 x_i の対応点 $h_f(x_i)$ を用いて、マーカ領域 Ω およびマーカ周辺領域 $\partial\Omega$ の特徴点以外の画素のオプティカルフローを補間し、それに基づき背景画像を変形する。ここでは、背景画像ではマーカの周辺領域 $\partial\Omega$ に属する画素も、現フレームではマーカ領域内に遷移する場合があるため、マーカ領域 Ω だけでなくその周辺 $\partial\Omega$ の特徴点以外の画素の動きも補間する。

ここでは、各画素は距離が近くかつ信頼度の高い特徴点と類似した動きをし、また隣接する画素同士も類似した動きをするという考えに基づき、以下のエネルギー関数を最小化することで、正対背景画像 B におけるマーカ領域 Ω およびマーカ周辺領域 $\partial\Omega$ 内の各画素の動きを推定する。

$$E = \sum_{i \in \Omega \cup \partial\Omega} \sum_{k \in K} \omega_{(i,k)} \|u_i - u_k\|^2 + \alpha \sum_{(i,j) \in A} \|u_i - u_j\|^2 \quad (5)$$

ただし、 K は特徴点の画素番号の集合、 A はマーカ領域 Ω かつマーカ周辺領域 $\partial\Omega$ において隣り合う画素番号の集合である。また、 u_i は、画素 i の正対背景画像 B に対するフレーム f での x 軸方向 y 軸方向の画素の変位ベクトルを表す。なお特徴点に関しては、処理 (i) で $u_k = h_f(x_k) - x_k$ と求められているため既知である。重み $\omega_{(i,k)}$ は、各画素と特徴点との距離および、特徴点の信頼度に基づき以下のように算出する。

$$\omega_{(i,k)} = \max_{k \in K} C(x_k) \exp\left(-\frac{\|x_i - x_k\|^2}{\sigma}\right) \quad (6)$$

式 (5) は、GPU を用いた共役勾配法により、高速に最小化することができる。

最後に、求めた各画素の動きに基づき、正対背景画像 B の各画素値を順投影しかつ線形補間を行うことで、正対背景画像 B の変形を行う。

4 実験

提案手法の有効性を示すために、図 5, 6, 7 に示す異なる 3 環境においてマーカを配置し実験を行った。それぞれ、形状が湾曲し格子状のテクスチャを持つシーン A, 形状が平面でストライプのテクスチャを持つシーン B, 形状はほぼ平面であるがしわのためにマーカが浮いており比較的ランダムなテクスチャを持つシーン C である。対象の AR マーカとして、図 4 に示す、一辺が 80mm のマーカが印刷された物体 (一辺 95mm, 厚み 7mm) を用い、カメラの位置姿勢推定には ARToolKit[1] を用いた。各パラメータは表 1 に示す値を用いた。背景画像の生成には画像修復 [3] を用いた。図 5, 6, 7(a) は入力画像、図 (b) は処理 (b-4) において背景画像変形を行わず、正対背景画像をその



図 4 実験で使ったマーカ

まま現フレームの正対画像のマーカ領域にコピーし光学補正を行った結果、図 (c) は提案手法の結果である。また、図 (d) は入力画像の正対画像とマーカ周辺での特徴点の追跡結果、図 (e) は特徴点の追跡に基づいてマーカ領域およびマーカ周辺領域の正対背景画像を変形した結果である。また、各図の 1 行目は初期フレームからカメラを動かしていない時点での結果、2 行目、3 行目はカメラの移動による初期フレームとは異なる視点での結果である。以下、各シーンでの結果について考察する。

図 5 に示す、湾曲した形状を持つシーン A では、図 (d) に示すようにカメラ位置によって正対画像上での周辺テクスチャの大きな見えの変化が確認できる。このため、画像変形なしに背景画像を重畳した結果 (図 (b)) では、マーカ周辺の境界においてテクスチャの大きなずれが生じている。一方、マーカ周辺の特徴点のトラッキングに基づき図 (e) に示すように画像を変形し、それを合成した提案手法による結果 (図 (c)) では、テクスチャのずれは見られず違和感のない結果が得られている。

図 6 に示す、形状が平面でストライプのテクスチャを持つシーン B では、図 (d) からは一見、正対画像上でテクスチャの見えの変化は見られないが、マーカの厚みのためにテクスチャの平行移動が生じている。このため、画像変形なしの結果 (図 (b)) では、直線のずれが生じている。一方、提案手法においては、図 (c) に示すように直線のずれは生じず、違和感のない結果が

表 1 実験におけるパラメータ

入力画像	640×480 画素
正対画像でのマーカサイズ	80×80 画素
マーカ領域 Ω	140×140 画素
マーカ周辺領域 $\partial\Omega$ の幅 l	15 画素
L_1, L_2	11 画素, 31 画素
N_2, N_3	80%, 1%
信頼度算出のための範囲 N	3×3 画素
探索範囲 G	5×5 画素
局所パッチ W のサイズ	11×11 画素
特徴点の相互影響の範囲 M	50×50 画素
κ, α, σ	0.001, 1000, 25

河合・佐藤・横矢：AR マーカ除去のための実時間背景画像変形

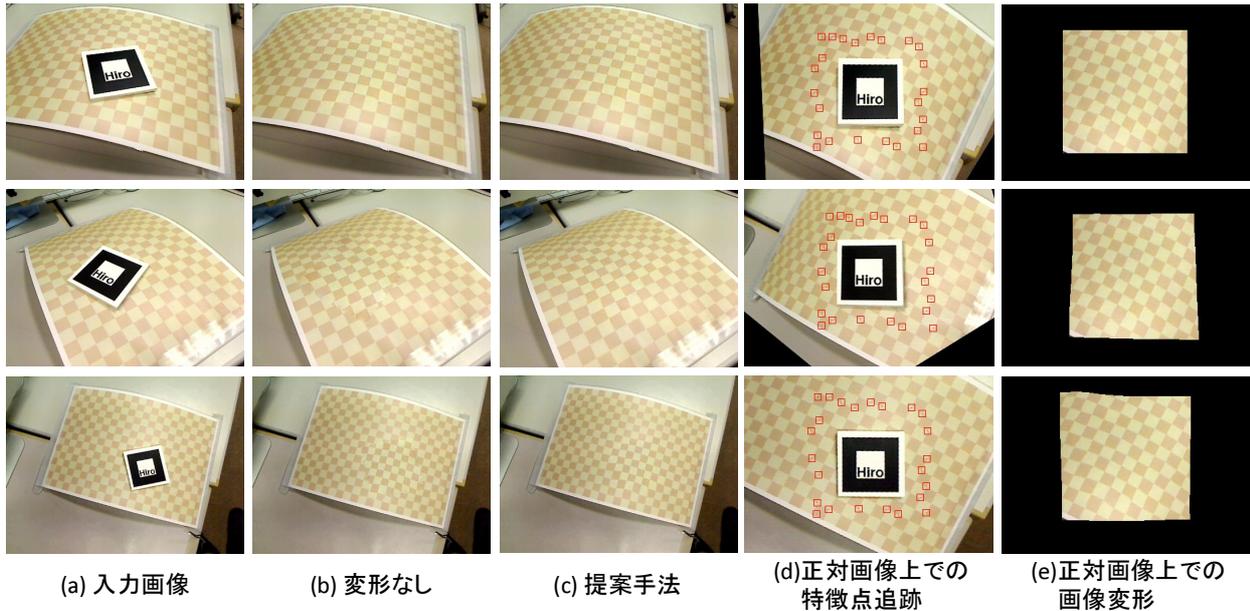


図5 シーン A(湾曲形状・格子状テクスチャ)の実験結果

得られている。ただし、図(d)に示すように、ストライプのパターンでは開口問題により必ずしも正対背景画像で検出された特徴点と同一の点と対応しているとは限らない。このため、図(e)に示すように背景が平面にも関わらず不自然な画像変形が生じている。しかし、エッジの方向とは異なる方向に対するオプティカルフローは求まるため、背景画像中のマーカ領域に周辺とは異なる模様が存在しない限りは、不自然な画像変形が生じていても違和感のない結果画像が得られる。

図7に示す、比較的ランダムなテクスチャパターンを持つシーンCでは、図(b)に示す画像変形なしの場合においてもテクスチャのずれは一見わかりにくい。しかし、提案手法の結果(図(c))と比較して、画像変形なしの場合には、不自然な色の変化が生じている。これは、テクスチャずれによって本来の正対背景画像には存在しないテクスチャの空間的な配置がマーカ領域の境界において生じてしまったために、処理(b-5)のポアソン画像合成を用いた色調補正において、不自然な色の伝搬が発生したためである。

なお、シーンA, B, Cにおいて検出された特徴点の数はそれぞれ、24, 20, 18であった。また処理速度はそれぞれ、約4.5fps, 4.2fps, 4.3fpsであった。

5 まとめ

本稿では、ARマーカの違和感のない視覚的除去を実現するための、背景画像の実時間変形手法を提案した。実験により、マーカに厚みがあり、マーカ背景の形状が湾曲している場合や多少の凹凸がある場合、またマーカ背景のテクスチャがテンプレートマッチング

により画像間で対応が付きやすい場合だけでなく、開口問題が生じるストライプ模様の場合においても画像変形により違和感のないマーカ除去を実現できることを確認した。しかし、現状の変形手法では隣り合う画素が同様の動きをするという前提に基づいているため、背景に急な形状変化がある場合には対応できない可能性が高い。このため、今後は様々な背景形状にも対応できる手法を開発する必要がある。また、マーカに限らず一般物体を除去する隠消現実感への応用を検討する。

謝辞

本研究の一部は、日本学術振興会科学研究費補助金、基盤研究(A), No. 23240024, 若手研究(B), No. 24700118による。

参考文献

- [1] H. Kato and M. Billinghurst: "Marker tracking and hmd calibration for a video-based augmented reality conferencing system," Proc. Int. Workshop on Augmented Reality, pp. 85-94, 1999.
- [2] S. Siltanen: "Texture generation over the marker area," Proc. Int. Symp. on Mixed and Augmented Reality, pp. 253-254, 2006.
- [3] N. Kawai, M. Yamasaki, T. Sato and N. Yokoya: "Diminished reality for AR marker hiding based on image inpainting with reflection of luminance changes," ITE Trans. on Media Technology and Applications, Vol. 1, No. 4, pp. 343-353, 2013.
- [4] F. I. Cosco, C. Garre, F. Bruno, M. Muzzupappa and M. A. Otaduy: "Augmented touch without visual obstruction," Proc. Int. Symp. on Mixed and Augmented Reality, pp. 99-102, 2009.

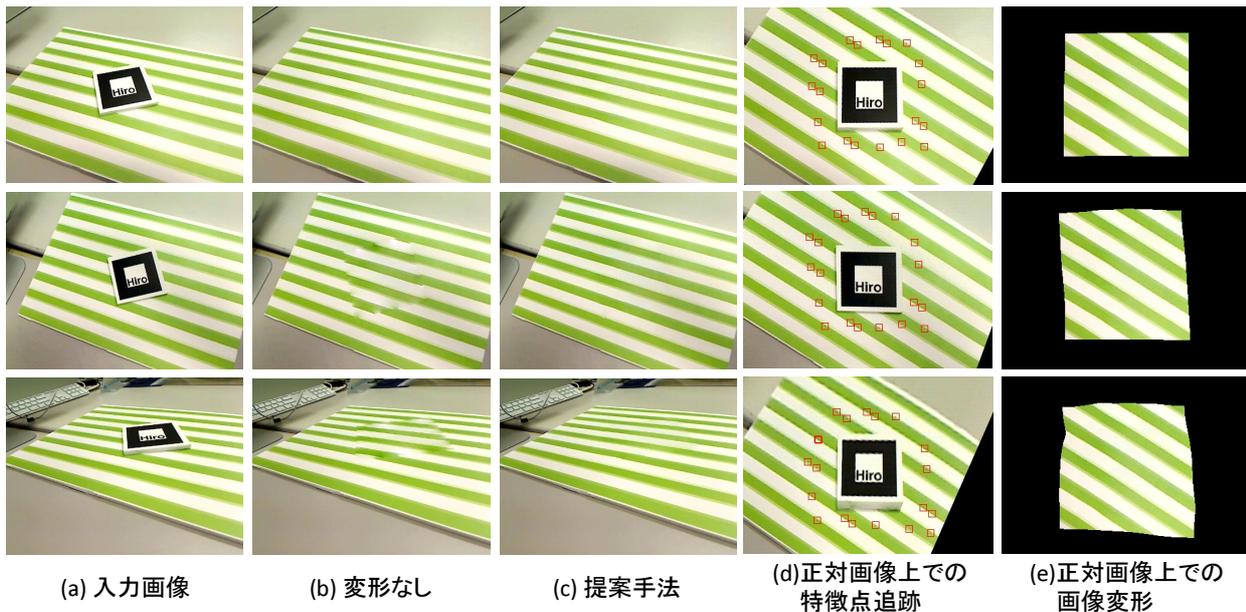


図6 シーンB(平面形状・ストライプテクスチャ)の実験結果

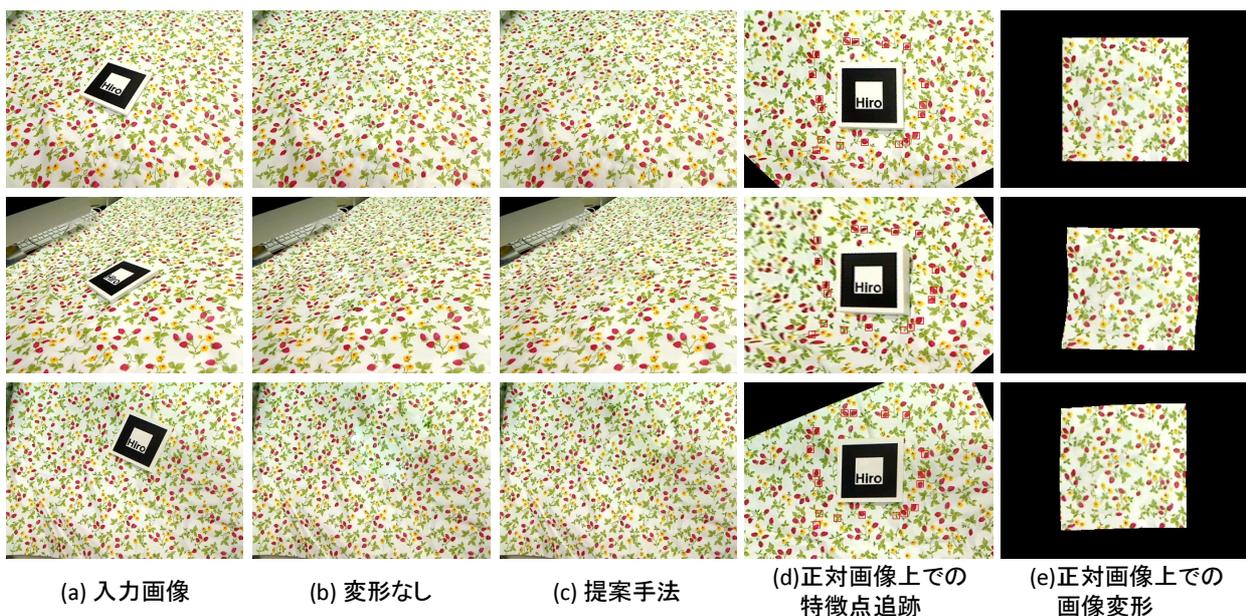


図7 シーンC(ほぼ平面だが凹凸のある形状・ランダムテクスチャ)の実験結果

- [5] 森尚平, 小向啓文, 柴田史久, 木村朝子, 田村秀行: “隠消現実感における隠背景平面拘束と周辺参照領域の効果的利用”, 日本バーチャルリアリティ学会論文誌, Vol. 19, No. 2, pp. 131–140, 2014.
- [6] Z. Li, Y. Wang, J. Guo, L.-F. Cheong and S. Z. Zhou: “Diminished reality using appearance and 3d geometry of internet photo collections,” Proc. Int. Symp. on Mixed and Augmented Reality, pp. 11–19, 2013.
- [7] J. Herling and W. Broll: “Advanced self-contained object removal for realizing real-time diminished reality in unconstrained environments,” Proc. Int. Symp. on Mixed and Augmented Reality, pp. 207–212, 2010.
- [8] J. Herling and W. Broll: “High-quality real-time video inpainting with pixmix,” IEEE Trans. Visualization and Computer Graphics, Vol. 20, No. 6, pp. 866–879, 2014.
- [9] N. Kawai, T. Sato and N. Yokoya: “Diminished reality considering background structures,” Proc. Int. Symp. on Mixed and Augmented Reality, pp. 259–260, 2013.
- [10] P. Pérez, M. Gangnet and A. Blake: “Poisson image editing,” ACM Trans. Graphics, Vol. 22, No. 3, pp. 313–318, 2003.