

プロジェクターカメラシステムによる移動非剛体に対する動的テクスチャリングに向けた基礎検討

中村 隆之^{†1} 小山田 雄仁^{†1}
斎藤 英雄^{†1} Julien Pilet^{†1}

Augmented Reality の分野で、現実物体表面に対し、仮想的な情報の表示をプロジェクターで行う研究が注目されている。PC のディスプレイや Head Mounted Display とは異なり、実在する物体の見え方を変える事が出来るという利点がある。本研究では、テクスチャを持つ非剛体に対して、任意の位置にプロジェクターで別のテクスチャを投影する事で、動的に非剛体上のテクスチャを変化させる動的テクスチャリングを目的とする。対象である非剛体自身が移動、変形をするため、非剛体のトラッキングを行う必要があるが、それに加えて、プロジェクターの投影によって非剛体の見え方は変化するため、問題はより困難になる。これら二つの問題を同時に解決するために、剛体のトラッキング手法として知られており、また近年では非剛体のトラッキングにも利用されている Lukas-Kanade 法を拡張した Modified Deformable Lukas-Kanade 法を提案する。本稿では、実環境への適用の予備実験として、シミュレーションで合成した画像に対して提案手法を適用し、その効果を示す。

Towards augmenting a moving and deforming surface by direct projection

TAKAYUKI NAKAMURA,^{†1} YUJI OYAMADA,^{†1}
HIDEO SAITO^{†1} and JULIEN PILET^{†1}

In the context of augmented reality, we aim to project patterns on a moving textured surface. Such a system can be used to augment a pattern printed on a real object. For example, it could animate a character on a T-Shirt. Since the surface is moving, it has to be tracked. However, the projection hinders tracking, since it changes the surface appearance. We propose to address this problem by modifying the Lucas-Kanade algorithm, a well known tracking method recently extended to handle deformable surfaces. Our simulated scenario consists of a projector and a camera. Our algorithm iteratively minimizes the difference between the captured image and the target image which combines the origi-

nal texture and the projection. Preliminary synthetic results demonstrate the efficiency of our approach applied to complex surfaces.

1. 本研究の背景

近年、光学式プロジェクターの研究開発が進んだ事により、明るさ、色の再現性、コントラスト比といった基本機能が優れたプロジェクターを、一般消費者が安価に手に入れる事が出来るようになった。その結果として、プロジェクターは一般的なディスプレイ装置の一つとなったと言える。プロジェクターは PC 等から送られた映像信号を変換した光を発する機能のみを持ち、自らは映像の結像を行わない点が他のディスプレイデバイスとは大きく異なる。投影された光は、何らかの結像装置 (一般的には白色スクリーン) 上で結像する事で初めて映像を表示する事になる。元来、テクスチャを持たない白色平面に対して、その光軸が垂直になるように投影する事を前提として設計されていたが、近年では、大半のプロジェクターにはキーストーン補正機能 (幾何学的ゆがみを補正) が搭載され、赤や緑と言った単色平面への投影を可能にするカラー補正機能が搭載されたモデルも販売されており、プロジェクターの使用環境に関する自由度は以前よりも上がっている。

このような背景から、仮想物体を現実世界に付加する技術である Augmented Reality (AR) の分野でプロジェクターの導入が注目されている。AR のシステムを構築するためには、現実世界に重畳表示したい仮想的な情報を、表示したい位置に違和感なく表示する事が重要になる。Computer Vision の技術に基づいた AR では、カメラで撮影した画像から撮影対象シーンの解析を行い、仮想的な情報をシーンに対して整合性を保つように重畳表示を行う。既存のシステムでは、仮想的な情報をコンピュータ上のディスプレイや Head Mounted Display にのみ表示させるものとなっている¹⁴⁾。プロジェクターを仮想的な情報を重畳表示させるためのデバイスとして用いる事が出来れば、現実世界の物体の見え方そのものを変える事が出来、より臨場感のあるシステムを期待出来る⁴⁾。

2. 関連研究

本章では、まず始めにプロジェクターを用いた AR について、次に AR の分野で行われて

^{†1} 慶應義塾大学大学院理工学研究科
Keio University

きた対象物体のトラッキングに関する研究について述べ、従来研究で対象とされた問題及び対象とされていない問題についてまとめる。

2.1 プロジェクタを用いた AR

プロジェクタを用いた AR の研究は、アプリケーションそのものに関する研究と、ディスプレイである被投影物体のトラッキングに関する研究の二つに大別する事が出来る。以下、各関連研究について述べていく。

2.1.1 プロジェクタを用いた AR アプリケーション

プロジェクタを用いた AR は Spatial Augmented Reality(SAR) と呼ばれる⁴⁾。SAR として最も有名な研究として、Shader Lamps が挙げられる^{2),16)}。Shader Lamps のコンセプトは、現実物体のアピアランスをプロジェクタの投影によって変える事にある。3次元形状が既知の物体をディスプレイとし、キャリブレーション済みのプロジェクタ-カメラペアを用いる事で、ディスプレイ物体に対して任意のテクスチャが正確に重なるように投影する事が出来る。被投影物体そのもののテクスチャと投影されたモデルのテクスチャがマージされる事で、被投影物体のアピアランスを変える事が出来る。

従来行われていた SAR のシステムは、被投影物体に関して

- 一枚もしくは複数の平面から構成される^{3),7),21)}
- 形状が既知の立体である¹⁶⁾

と制約を加える事で、プロジェクタが投影する仮想的な情報を、対象シーンに対して整合性を持たせて表示を行う。

2.1.2 プロジェクタの被投影物体のトラッキング

一方で、プロジェクタで投影する仮想的な情報を、対象シーンに対して整合性を持たせて表示する事を目的とした研究が行われている。これらの研究は、被投影物体上で表示された映像上に生じる歪みの種類によって、幾何学補正、光学補正、焦点補正とそれぞれ研究されているが、ここでは幾何学補正を被投影物体のトラッキングと呼び、以降はこの被投影物体のトラッキングについてのみ扱う。

被投影物体のトラッキングをオンラインで行う方法は大きく分けて二つある。一つは人間が認識できないキャリブレーションパターンを投影画像に埋め込む方法である^{5),6)}。カメラで撮影した画像を時系列に見ていく事で、人間が認識出来ないパターンを検出し、それを位置検出の計算に利用する。不可視パターンを埋め込むため、プロジェクタのダイナミックレンジが下がってしまうという欠点はあるが、パターンの検出や検出後の計算は簡単な問題となる。もう一つの方法は、画像そのものから歪みに関する情報を計算する方法であ

る^{9),15),20)}。これは、投影する映像に対して何らかの処理を加える必要はないが、カメラで撮影した画像からどの程度の情報が得られるかに依存する。また、上記二つの方法を組み合わせた方法もある^{19),23)}。

2.2 対象物体のトラッキング

次に、プロジェクタに限らず AR の分野において行われてきた対象物体のトラッキングに関する研究について述べる。AR におけるトラッキングは対象物体の性質によって、剛体トラッキングと非剛体トラッキングの二つに分類できる。剛体のトラッキングで最も有名な手法の一つである Lucas-Kanade(LK) 法は、剛体の変形(回転、並進など)前後のアピアランスの変化を比較することにより、剛体の移動を計算し、マッチングを行う手法である¹⁾。剛体のトラッキングに関しては、ロボスタかつリアルタイムに実現可能な手法が数多く提案されている^{10)-12),17)}。

非剛体表面のトラッキングは、大きく分けて二つに分類することが出来る。一つは、特徴点を抽出し、その特徴点の動きをトラッキングする方法である¹⁴⁾。既知のテクスチャを持つ非剛体を対象とし、変形前後の特徴点の見えの変化をトラッキングに利用している。もう一つは、画像全体のアピアランス(見え)の変化を検出し、トラッキングする方法である²²⁾。他にも、オクルージョン⁸⁾、物体の伸び¹³⁾、テクスチャの少ない対象¹⁸⁾などに特化した非剛体表面のトラッキングの研究が行われている。本手法は、非剛体を微小な剛体の集合であると仮定し、LK 法を拡張した Deformable Lucas-Kanade 法をさらに拡張したものである。

2.3 本研究の位置づけ

2.1 章で述べたように、今まで行われてきた SAR の研究では被投影物体が剛体に限られていたため、アプリケーションとしてはどうしても制約が強いものになってしまう。対象物体のトラッキングという問題に話を限定すると、2.2 章で述べたように、非剛体の移動や形状変化を推定する研究は近年盛んに行われており、オンラインで非剛体の移動や形状変化を扱う事自体は可能である。この事から、非剛体トラッキングの分野で行われている考えに基づいてディスプレイ物体を扱う事で、SAR において非剛体をディスプレイ物体とする事は実現可能であると期待出来る。

3. 目的とシステム

本研究の最終目標は、非剛体表面へのプロジェクタによる動的テクスチャリングである。つまり、非剛体表面の変形、プロジェクタ画像の重畳によるテクスチャの変形、これらを同時にトラッキングすることを目的とする。システムについて、本研究の目標であるプロジェ

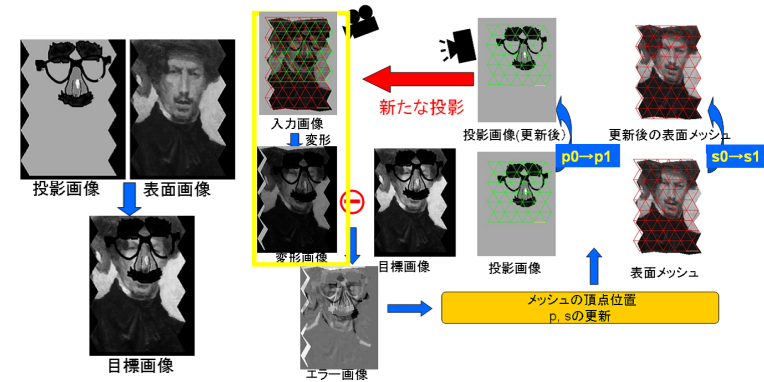


図 1 プロジェクタを用いた非剛体への動的テクスチャリング

クタを用いた非剛体への動的テクスチャリングの概要を図 1 に示す。システムは、図 1(a) に示すようにプロジェクタ、カメラそして対象物体である非剛体で構成される。あらかじめ、図 1(b) に示すように、非剛体物体の初期状態(歪みが生じていない状態の見え)を撮影し、プロジェクタで投影するテクスチャ(図 1(c))を用意する。システムが稼動している時、任意に動く非剛体をカメラで撮影し、非剛体の見えの変化から非剛体表面がどのように変形しているかを計算する。次に、変形した非剛体表面にフィッティングするように変形したテクスチャを投影する事で、非剛体表面に対してテクスチャリングを行う。以降、時々刻々と変形する非剛体に対してこの処理を繰り返す事で、動的テクスチャリングを実現する。その際に、非剛体の変形だけではなく、プロジェクタで投影するテクスチャも非剛体表面の見えを変化させるため、このプロジェクタの投影するテクスチャによる見えの変化も同様に考慮に入れる。本稿では、手法の提案と、トラッキングの精度、有効性を調べるためのシミュレーションを行う。

4. 提案手法

本手法は、勾配法に基いて画像上の変化を解析することでトラッキングを行う Lucas-Kanade 法 (LK 法)¹⁾ を基盤とする。LK 法は、剛体のトラッキングを目的に提案された手法だが、近年では非剛体のトラッキング法 (Deformable Lucas-Kanade 法²²⁾) として適用可能である事が証明されており、非剛体を被投影物体とする動的テクスチャリングに対して



(a) オフライン

(b) オンライン

図 2 提案手法のフロー

も効果が期待できる。しかしながら、動的テクスチャリングでは、被投影物体自身の変形による見えの変化に加え、プロジェクタの投影も見えの変化の原因となるため、LK 法をそのまま適用する事は出来ない。本章では、プロジェクタの投影による見えの変化と非剛体の変形による見えの変化を同時に扱う Modified Deformable Lucas-Kanade 法 (MDLK 法) を提案する。

本手法は、オフライン、オンラインの二段階の処理で構成される。オフライン時では、画像の登録と処理に時間がかかる行列計算を行う (図 2(a))。まず始めに、対象物体である非剛体の初期状態 (歪みが生じていない状態) を撮影し、プロジェクタで投影するテクスチャを用意する。これら変形の無い参照画像をテンプレートと呼ぶ。また、処理を通して一定の値をとる行列計算を行う。オンライン時では、プロジェクタでテクスチャリングされている非剛体表面を撮影し、非剛体の変形による見えの変化とテクスチャリングによる見えの変化を計算する (図 2(b))。オフライン時に処理に時間がかかる行列計算を行っているため、見えの変化を高速に計算することが出来る。

4.1 Lucas-Kanade 法

LK 法は、テンプレートと入力画像の差を計算し、その差を減らすようにテンプレートを繰り返し処理によって変形する事で変形パラメータを計算する手法である¹⁾。繰り返し処理

の度に Hessian の計算といった行列演算を行うため、オンラインアプリケーションに不向きであるといった問題や、剛体を対象として設計されたため、複雑な形状変化をする非剛体に対して適用できないといった問題を持つ。

4.2 Modified Deformable Lucas-Kanade 法

MDLK 法では、非投影物体表面のテクスチャとプロジェクタが投影するテクスチャを同時に扱う。そこで、表面テクスチャの変形を表す変数を s 、投影テクスチャの変形を表す変数を p と表現することとする。 s, p は、各メッシュの頂点の集合を意味し、この s, p を用い見えの変化を計算する。表面テクスチャと投影テクスチャそれぞれをテンプレート画像 $T(x)(x=(x, y))$ として保持されているとする。それぞれのテンプレートを、表面画像 T_s 、投影画像 T_p とし、表面画像の任意の位置に投影画像を配置した目標の見えとなる目標画像 T を作成しておく。MDLK 法は非剛体の形状と投影画像の位置を、2つの三角メッシュで表すことにより、画像全体を、三角パッチに分けた局所領域の変形の集合として表現している。この 2D メッシュは N 個の各頂点が六角形につながった形状を持っている。画像を微小なパッチの集合として表現するため、メッシュの頂点座標を表す z_2 を表面の変形パラメータ s と投影の変形パラメータ p を列ベクトルとして持つ行列として定義する。

$$z_2 = \begin{bmatrix} s_x & s_y & p_x & p_y \\ x_{s_1} & \dots & x_{s_N} & y_{s_1} & \dots & y_{s_N} & x_{p_1} & \dots & x_{p_N} & y_{p_1} & \dots & y_{p_N} \end{bmatrix}^T \quad (1)$$

今、三角メッシュの中のある一点 x を考える。 $i, j, k \in [1, N]$ とすると、三角形の各頂点の座標は $(x_i, y_i), (x_j, y_j), (x_k, y_k)$ である。変形後の三角パッチから対応する三角パッチへの画素のマッピングは、変形関数 $W(x, z_2)$ を用いて

$$W(x, z_2) = \begin{bmatrix} x_i & x_j & x_k \\ y_i & y_j & y_k \end{bmatrix} \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix}^T \quad (2)$$

となる。ここで、 (b_1, b_2, b_3) は点 x における重心座標である。これらの三角メッシュを用いて、非剛体のトラッキングを行う。

前述したように、LK 法は Hessian の計算を毎フレーム行うため処理に時間がかかる。この問題を解決するために、MDLK 法では、LK 法におけるテンプレートと入力画像の役割を交換することで、処理に時間のかかる Hessian の計算をオフラインで一度計算すればよいものとする。MDLK 法は、メッシュの頂点 z_2 を直接変形パラメータとして使用しているた

め、 Δz_2 をメッシュの頂点の変形量として定義出来る。入力画像 $I(x)$ を変形関数 W 及び変形パラメータ z_2 を用いて変形した画像は $I(W(x, z_2))$ と表すことが出来る。よって、次式に示すように、見えの変化から生じる残差を、appearance エネルギー E_a をとして定義できる。

$$E_a(z_2) = \sum_x [T_s(W(x, \Delta z_2)) + T_p(W(x, \Delta z_2)) - I(W(x, z_2))]^2 \quad (3)$$

一般的に、非剛体表面の形状、及び投影画像の変形は、3次元の情報をそれぞれ $2N$ 個の変数を持つ 2次元メッシュで近似するため、不良設定問題である。この問題を解決するために、変形するメッシュの形状を保つ regularization エネルギー E_r を導入する。

$$E_r(z_2) = z_2^T K'' z_2 \quad (4)$$

ここで、行列 $K'' \in R^{4N \times 4N}$ は、 $2N \times 2N$ 行列 K で構成される行列を表す。

$$K'' = \begin{bmatrix} K & 0 \\ 0 & K \end{bmatrix}, K = \begin{bmatrix} K'^T K' & 0 \\ 0 & K'^T K' \end{bmatrix} \quad (5)$$

K' は、 (i, j, k) 毎に一行、メッシュの頂点毎に一行の行列であり、 i, j, k に対応する位置では $-1, 2, -1$ 、それ以外が 0 で満たされた sparse and banded matrix となる。

これら 2つのエネルギーの和 E_A を最小化する事で、変形パラメータを推定できる。 λ_r を正規化係数として、

$$E_A(z_2) = E_a(z_2) + \lambda_r E_r(z_2) \quad (6)$$

式 6 を一次テイラー展開すると、

$$\sum_x [T_s(W(x, z_{20})) + \nabla T_s \frac{\partial W}{\partial z_2} \Delta z_2 + T_p(W(x, z_{20})) + \nabla T_p \frac{\partial W}{\partial z_2} \Delta z_2 - I(W(x, z_2))]^2 + \lambda_r (z_2 + \Delta z_2)^T K'' (z_2 + \Delta z_2) \quad (7)$$

が得られる。 $\nabla T_s, \nabla T_p$ は $W(x, z_{20})$ によって変形された各テンプレートの勾配、 z_{20} は図 3 に示す参照メッシュであり、無変形であるテンプレートの状態を表す。 $W(x, z_{20})$ が項等な変形なので、 $T(W(x, z_{20})) = T(x)$ とみなせる。 Δz_2 によって式 7 を偏微分すると、

$$2 \sum_x [\nabla T_s \frac{\partial W}{\partial z_2} + \nabla T_p \frac{\partial W}{\partial z_2}]^T [T_s(x) + \nabla T_s \frac{\partial W}{\partial z_2} \Delta z_2 + T_p(x) + \nabla T_p \frac{\partial W}{\partial z_2} \Delta z_2 - I(W(x, z_2))]^2 + \lambda_r K'' (z_2 + \Delta z_2) \quad (8)$$

となる。式 7 が示すエネルギーを最小化する Δz_2 は、式 8 を 0 とする時の Δz_2 であるため、

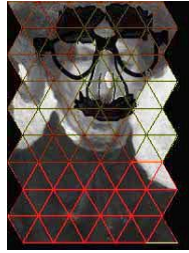


図 3 テンプレートの参照メッシュ

$$\Delta z_2 = H_4^{-1} \sum_x \left[\nabla T_s \frac{\partial W}{\partial z_2} + \nabla T_p \frac{\partial W}{\partial z_2} \right]^T [I(W(x, z_2)) - T_s(x) - T_p(x)] - \lambda_r H_4^{-1} K'' z_2 \quad (9)$$

が導出できる。ここで H_4 は $4N \times 4N$ の Hessian である。

$$H_4 = \sum_x \left[\nabla T_s \frac{\partial W}{\partial z_2} + \nabla T_p \frac{\partial W}{\partial z_2} \right]^T \left[\nabla T_s \frac{\partial W}{\partial z_2} + \nabla T_p \frac{\partial W}{\partial z_2} \right] + \lambda_r K'' \quad (10)$$

以上より、Hessian H_4 は変数 z_2 に独立であり、最適化の反復中も一定であるため、オフラインで計算を済ませる事が可能である事が分かる。よって、メッシュの頂点 z_2 が効率的に計算できる。また、 z_2 は変形パラメータとして使用されているため、更新は次式のように直接計算することができる。

$$z_2 \leftarrow z_2 - \Delta z_2 \quad (11)$$

MDLK 法の流れをまとめると、以下ようになる。

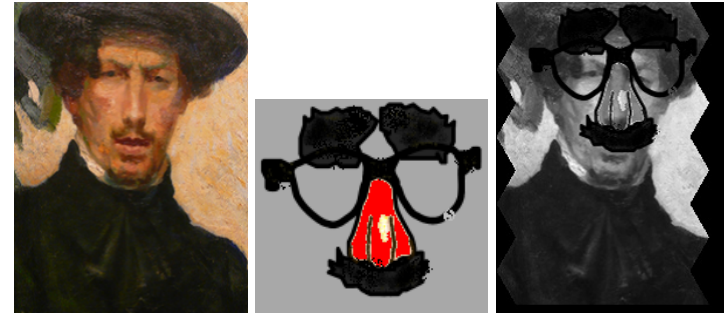
- (1) 表面テクスチャ T_s と投影テクスチャ T_p からテンプレート画像 T を生成。
- (2) テンプレート画像に対して三角メッシュを張った参照メッシュ z_{20} を生成。
- (3) 与えられた入力画像 I に対して三角メッシュ z_2 を張る。
- (4) 式 7 を最小化する Δz_2 を求める。
- (5) z_2 を式 11 を用いて更新。
- (6) 収束するまで、4,5 の処理を繰り返す。

5. 実験と結果

本章では、提案手法によるプロジェクタを用いた動的テクスチャリングのシミュレーション実験を行う。非剛体表面を表す表面テクスチャ(図 4(a))とプロジェクタで投影するテク

スチャ(図 4(b))を用意し、二つの画像を合成したテンプレート画像(図 4(c))を生成した。実験環境は以下の通りである。

- 表面画像 : 199×265 [pix]
- 投影画像 : 152×139 [pix]
- メッシュサイズ : 33.584 [pix]
- CPU : Intel Core2 Duo CPU E7400 @ 2.80GHz
- メモリ : 3.5GB
- $\lambda_r = 8 \times 10^5$



(a) 表面テクスチャ (b) 投影テクスチャ (c) テンプレート画像

図 4 シミュレーション実験で使った画像

シミュレーションで想定する非剛体の動きは、メッシュのパッチサイズ 33.584pix の約 1/3 である 10pix とした。入力画像に張る三角メッシュ z_2 の初期値は、参照メッシュ z_{20} をずらしたものとし、表面テクスチャ上のメッシュを構成する頂点の位置を以下のように変化させ、提案手法の挙動を確認した。

- (1) 平行移動 (剛体): ランダムに 0~10pix 平行移動
- (2) 回転 (剛体): 画像を縮小した後に 90 度回転させ、その後ランダムに 0~5pix 平行移動
- (3) ランダム (非剛体): 各頂点をランダムに 0~5pix 移動させ、その後ランダムに 0~10pix 平行移動

図5に、繰り返し処理に対して、テンプレート画像 T と入力画像を変形させた $I(W(x, z_2))$ の Root Mean Square Error (RMSE) の推移を示す。iteration の回数を重ねるにつれ、徐々に RMSE の値が減り、およそ 100 回程度で収束していることが分かる。

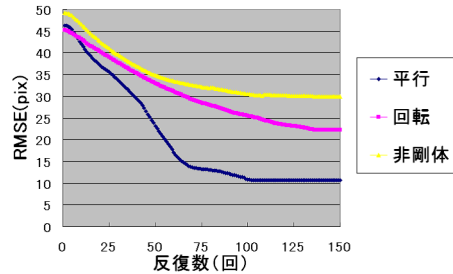


図5 繰り返しによる RMSE の推移

表5に、収束時での RMSE の値を示す。RMSE の平均値をみると、平行移動の結果が最も RMSE が小さく、非剛体の動きを表すランダムに動かした時の結果が最も RMSE が大きくなっている。一方で、標準偏差の値を見ると、ランダムなケースが最も安定しており、回転をした時の値が最大となっている。画像を回転させる際に、画像を縮小したため、局所解に陥ったと考えられる。

表1 収束時における RMSE

	平行移動	回転	ランダム
平均 (pix)	10.6	22.3	29.7
標準偏差 (pix)	2.5	11.2	1.0

収束に要した処理速度は、平行移動、回転、ランダムでそれぞれ、11.5fps, 12.7fps, 12.7fps となり、およそ 12fps の処理速度が得られた。図5からも、非剛体の動きによらず、一定の処理速度が得られると言える。

図6に、変形画像 $I(W(x, z_2))$ が初期状態と収束時でどのように変化するかを示す。テンプレート画像と変形画像の残差画像から、初期状態に比べ収束時の方がエラーが少なくなっている事が分かる。これから、提案した MDLK 法によって、動的テクスチャリングに必要な非剛体のトラッキングをプロジェクタの投影下でも行えていることがわかる。

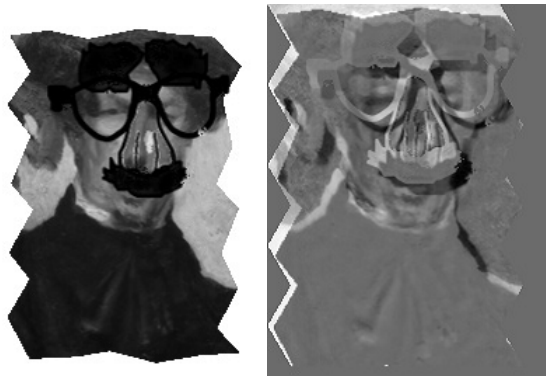
6. 結 論

本論文では、テクスチャをもつ非剛体表面に対し、任意の位置に投影を行うための動的テクスチャリングを目的とした。動的テクスチャリングを実現するためには、非剛体表面の形状変化による見えの変化と、投影による見えの変化を同時に扱う必要がある。この問題を解決するために、勾配方向に解を繰り返し求める LK 法を非剛体の形状変化に対応させた DLK 法を拡張した Modified Deformable Lucas-Kanade 法 (MDLK 法) を提案した。DLK 法の Hessian を拡張することで、形状変化とプロジェクタそれぞれの変形のパラメータを同時に扱うことを可能とした。実験では、提案手法によるプロジェクタを用いた動的テクスチャリングのシミュレーションを行い、テクスチャを持つ非剛体の変形、移動による見えの変化とプロジェクタの投影による見えの変化が同時に発生した状況でも非剛体表面のトラッキングができることを示した。今後、本手法を GPU を利用した並列処理を行うことによって高速化し、実際にプロジェクタを使用した実環境で実験を行う。

参 考 文 献

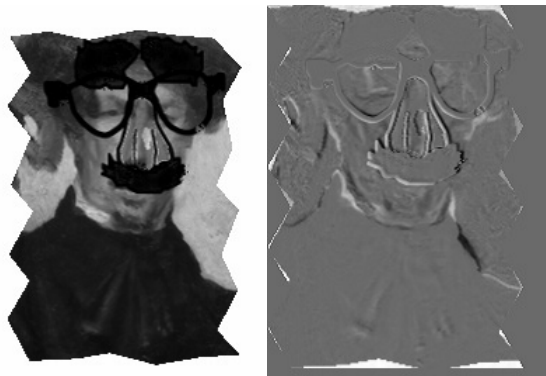
- 1) SIMON BAKER and IAIN MATTHEWS. Lucas-kanade 20 years on: A unifying framework. *International Journal of Computer Vision* 56(3), 221-255, 2004, pp. 221-255, 2003.
- 2) Deepak Bandyopadhyay, Ramesh Raskar, and Henry Fuchs. Dynamic shader lamps : Painting on movable objects. In *IEEE and ACM International Symposium on Augmented Reality*, pp. 207-216, 2001.
- 3) François Béard. The magic table: Computer-vision based augmentation of a whiteboard for creative meetings. In *International Workshop on Projector-Camera Systems (PROCAMS)*, 2003.
- 4) Oliver Bimber and Ramesh Raskar. *Spatial Augmented Reality: Merging Real and Virtual Worlds*. A K Peters LTD, 2005.
- 5) Daniel Cotting, Martin Naef, Markus Gross, and Henry Fuchs. Embedding imperceptible patterns into projected images for simultaneous acquisition and display. In *IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 100-109, 2004.
- 6) Daniel Cotting, Remo Ziegler, Markus Gross, and Henry Fuchs. Adaptive instant displays: Continuously calibrated projections using per-pixel light control. *Computer Graphics Forum*, Vol.24, No.3, pp. 705-714, 2005.
- 7) Shilpi Gupta and Christopher Jaynes. The universal media book: Tracking and augmenting moving surfaces with projected information. In *International Sympo-*

- sium on Mixed and Augmented Reality (ISMAR)*, pp. 177–180, 2006.
- 8) A. Hilsman and P. Eisert. Tracking and Retexturing Cloth for Real-Time Virtual Clothing Applications. *Computer Vision/Computer Graphics Collaboration Techniques*, p.94, 2009.
 - 9) Tyler Johnson and Henry Fuchs. Real-time projector tracking on complex geometry using ordinary imagery. In *IEEE International Workshop on Projector-Camera Systems (PROCAMS)*, 2007.
 - 10) V. Lepetit and P. Fua. *Monocular model-based 3d tracking of rigid objects*. Now Publishers Inc, 2005.
 - 11) V. Lepetit, P. Laguerre, and P. Fua. Randomized trees for real-time keypoint recognition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005*, Vol.2, 2005.
 - 12) D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, Vol.60, No.2, pp. 91–110, 2004.
 - 13) Francesc Moreno-Noguer, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua. Capturing 3d stretchable surfaces from single images in closed form. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
 - 14) J. Pilet, V. Lepetit, and P. Fua. Fast non-rigid surface detection, registration and realistic augmentation. *International Journal of Computer Vision*, Vol.76, No.2, pp. 109–122, 2008.
 - 15) Ramesh Raskar, Jeroen van Baar, Paul Beardsley, Thomas Willwacher, Srinivas Rao, and Clifton Forlines. ilamps: geometrically aware and self-configuring projectors. *ACM Transactions on Graphics*, Vol.22, pp. 809–818, 2003.
 - 16) Ramesh Raskar, Greg Welch, Kok Lim Low, and Deepak Bandyopadhyay. Shader lamps: Animating real objects with image-based illumination. In *Eurographics Workshop on Rendering*, pp. 89–102, 2001.
 - 17) E. Rosten and T. Drummond. Fusing points and lines for high performance tracking. In *Tenth IEEE International Conference on Computer Vision, 2005. ICCV 2005*, Vol.2, 2005.
 - 18) M. Salzmann, R. Urtasun, and P. Fua. Local deformation models for monocular 3d shape recovery. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, June 2008.
 - 19) Michael Waschbüsch, Stephan Würmlin, Daniel Cotting, Filip Sadlo, and Markus Gross. Scalable 3d video of dynamic scenes. *The Visual Computer*, Vol.21, pp. 629–638, 2005.
 - 20) Ruigang Yang and Greg Welch. Automatic and continuous projector display surface calibration using every-day imagery. In *International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, 2001.
 - 21) Tatsuo Yotsukura, Shigeo Morishima, Frank Nielsen, Kim Binsted, and Claudio S. Pinhanez. Hypermask - projecting a talking head onto a real object. *The Visual Computer*, Vol.18, No.2, pp. 111–120, April 2002.
 - 22) Jianke Zhu, Michael R. Lyu, and Thomas S. Huang. A fast 2d shape recovery approach by fusing features and appearance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol.31, No.7, pp. 1210–1224, July 2009.
 - 23) Stefanie Zollmann, Tobias Langlotz, and Oliver Bimber. Passive-active geometric calibration for view-dependent projections onto arbitrary surfaces. *Journal of Virtual Reality and Broadcasting*, Vol.4, No.6, pp. 1–11, 2007.



(a) 初期状態での $I(W(x, z_2))$

(b) 初期状態での残差画像



(c) 収束時での $I(W(x, z_2))$

(d) 収束時での残差画像

図 6 ランダム実験における残差の変化