

フレーム間の連続性を保持する動画編集法

谷田川 達也^{1,a)} 山口 泰^{1,b)}

概要: 本研究では、動画におけるフレーム間の連続性を保持しながら、動画の色調やテクスチャなどを編集するための手法を提案する。多くの動画処理においてオプティカル・フローなどといった位置の対応関係を用いて、あるフレームの情報を異なるフレームへと移送する手法が用いられている。その一方でこのような位置の対応関係を密に計算するためには多くの計算量を要するほか、遮蔽などの影響を受けやすく誤った対応関係がってしまうことがしばしばある。そこで本研究では、位置の対応関係を用いず、フレーム間の対応関係を一つの正方行列により記述する。これは、あるフレームの各ピクセルが持つ性質は、前のフレームのピクセルが持つ性質の線形結合により表現できるという仮定に基づいている。つまりこの線形結合の係数により構成される正方行列によってフレーム間の対応関係を記述する。本稿ではこの行列を「継承行列 (inheritance matrix)」と呼ぶ。本研究では動画の特定のフレームに対して施した編集結果それ自体を移送するのではなく、ピクセルごとに割り当てられる編集パラメータを編集マップの形で移送する。継承行列により編集マップを移送することで動画全体にユーザの意図に沿った編集をより少ないユーザ入力で実現する。本稿では、継承行列による連続性を保持した動画編集の結果に加え、継承行列の様々な動画処理手法への応用例も示す。

1. はじめに

近年、携帯電話やデジタルカメラなどの比較的安価なデバイスを用いて動画が撮影できるようになったことで、これまで一部の映像制作においてしか用いられてこなかった動画の編集技術に対する一般利用者からの要望が高まりつつある。静止画と異なり、多くのフレームから構成される動画は編集には多くの手間や時間を要する。こういった動画の編集は、静止画の編集と同様に、部分的なマスクを作成し、そのマスク領域内を順々に編集することで行われることが多い。ところが、これらのマスクの作成は、編集を行おうとする領域の輪郭が複雑であったり、曖昧であったりする場合に高度な技術を要する。また領域の境界にあたる部分の編集がどのように合成されるべきかなどはユーザの経験やその場の判断に基づくものが多く、日常的に動画編集を行わない一般利用者にとっては一層難しさが増す。

その中で、少ないユーザの入力情報から静止画全体を編集する研究が紹介された [9, 10]。これらの研究は、ユーザが画像の一部にストロークの形で入力した編集パラメータを元にして、静止画の全てのピクセルに編集パラメータを割り当てる。これらの手法の特徴は、割り当てられた編集パラメータが静止画中の物体領域をマスクするだけでな

く、領域の境界における合成の程度も同時に計算できるという点にある。この考え方は動画においても有用と考えられるが、これを動画に適応した例はそれほど多くはない。

これにはいくつかの理由があると考えられる。まず動画は多くのフレームから構成されるが故に、既存手法をそのまま適用しようとする場合には多くのユーザ入力が必要である。これはあまり現実的でない。次に、特定のフレームに施されたユーザ入力を元にして、ユーザの意図をフレーム間で次々に移送していくという方法が考えられる。移送の方法はいくつか考えられるものの、多くの動画処理においてはオプティカル・フローなどの位置の対応関係を用いるのが一般的である。実際、オプティカル・フローを用いて、領域分割のラベルを移送する研究例が紹介されている [8]。この手法は非常に応用範囲の広い手法ではあるが、オプティカル・フローの対応付けが不完全であったり、対応点のペアが十分に取れない場合には、移送そのものが失敗してしまうという問題点が指摘されている。

そこで本研究では、このような位置の対応関係に基づかないフレーム間の編集パラメータの移送法を提案する。これにより位置の対応関係を用いた処理が抱えていた問題の解決を目指すとともに、ユーザが一部のフレームに対して入力した編集パラメータの安定な移送を実現する。

¹ 東京大学 / JST CREST

^{a)} tatsuya@graco.c.u-tokyo.ac.jp

^{b)} yama@graco.c.u-tokyo.ac.jp

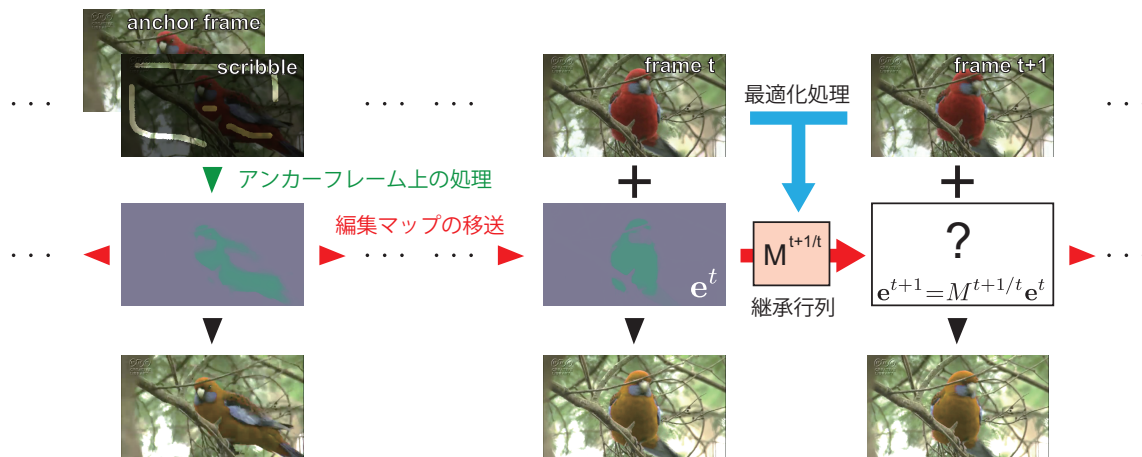


図 1 提案手法の概念図.

2. 関連研究

2.1 動画処理と情報の移送

動画を扱う処理の多くはオプティカル・フローなどの位置の対応関係を用いることが多い。一例として、動画の手ブレを除去する処理 [11] や動画のセグメンテーション [3] などは位置の対応関係を利用している。本研究で目的とする編集マップの移送にも位置の対応関係を使うことは可能である。エッジ保存型平滑化フィルタ (edge-preserving filter) に基づき近似的なオプティカル・フローを求める手法 [8] において、領域分割のラベルを移送する応用例が紹介されている。ところが、このような位置の対応関係にはいくつかの問題がある。まず、フレームの全てのピクセルに位置の対応関係を正確に与えるのは依然として時間を要するということが挙げられる [15]。また、遮蔽などの影響で全てのピクセルに対応関係を与えることが本来できない場合もある。そのため、対応のつかない場所に情報を与える目的でエッジ保存型平滑化フィルタなどが用いられるのだが、対応付けの失敗や対応関係の不足により、移送に失敗することがある [8]。そこで本研究では、編集マップの移送に位置の対応関係を用いない処理を導入する。

2.2 静止画の外観編集

以前から、静止画の外観、すなわち色調や露光、テクスチャなどを編集するための技術は多く研究がなされている。初期の HDR 画像のトーン・マッピング [5] や色転写 [12] (color transfer) は静止画全体を一定のルールに基づいて編集していたため、処理が高速である一方、意図しない編集結果となる場合があるなどの問題を抱えていた。近年では、ユーザの入力を元にして静止画を編集する手法が数多く提案されており、例えば、ユーザ制御によるモノクロ画像の着色 [9] やトーン・マッピング [10]、色調補正 [1]、色転写 [2] などが挙げられる。これらの処理に一貫するアイ

ディアとして、ユーザの入力を基にした最適化問題を解くことで、ユーザ入力に潜む意図を静止画全体に適用するという点が挙げられる。

このアイデアは動画に対しても有用であり、ユーザ入力からフレームに編集を施すことは可能であるが、動画の場合には全てのフレームにユーザ入力を行うというのは現実的ではない。そこで本研究では、ユーザが入力を行うフレームをアンカーフレームと呼び、最初にアンカーフレーム上の編集情報を作成する。この編集情報を以下では編集マップと呼ぶ。編集マップを逐次、隣接フレームに移送していくことで、動画全体に編集情報の付与を行う。

3. フレーム間連続性を保持する動画編集法

本研究の提案する手法は大きく二つの処理に分けられる。一つ目にユーザが入力を行ったフレーム上での処理である。このユーザが入力を行うフレームを本稿ではアンカーフレームと呼ぶ。静止画上でユーザ入力から画像全体の編集マップを推定する手法はすでにいくつか提案されており、本研究では An と Pellacini による外観編集法 [1] を用いる。二つ目の処理はアンカーフレームに対して作成された編集マップを隣接するフレームに移送するという処理である。この処理は隣接する二枚の画像から計算される継承行列を用いて行われる。継承行列はエネルギーの最適化により求められる行列で移送するマップの種類や値に依存しないという特徴を持つ。

本節は以下の流れに沿って提案手法の説明を行う。まず 3.1 項では、アンカーフレーム上での処理を説明する。次に 3.2 項でフレーム間の編集マップの移送について説明する。その後、3.3 項で継承行列の近似的計算法について説明し、最後に 3.4 項で動画編集の結果を紹介する。また、提案手法の概念図を図 1 に示す。

3.1 アンカーフレームにおける編集マップの作成

動画の各フレームが n ピクセルから成るとすると、各ピ

クセルに対して $i = 1, \dots, n$ という番号を与えることができる。ここで、ユーザによりピクセル i に与えられる編集パラメータを \hat{g}_i とおく。またユーザ入力が行われているかを表すラベルを \hat{w}_i とおく。具体的には \hat{w}_i はピクセル i に入力が行われているとき 1, 行われていないとき 0 をとる。最終的にピクセル i に与えられる編集パラメータ \hat{e}_i を求めるためのエネルギー関数 \hat{E} を次のように定義することができる。

$$\hat{E} = \sum_{i=1}^n \sum_{j=1}^n \hat{w}_i (\hat{e}_i - \hat{g}_j)^2 + \lambda \sum_{i=1}^n \sum_{j=1}^n \hat{z}_{ij} (\hat{e}_i - \hat{e}_j)^2 \quad (1)$$

この式において \hat{z}_{ij} はピクセル i と j の間の類似度を表し、次のように定義される。

$$\hat{z}_{ij} = \exp(-\|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j\|/(2\sigma_s^2)) \exp(-\|\hat{\mathbf{c}}_i - \hat{\mathbf{c}}_j\|/(2\sigma_c^2))$$

ここで、 $\hat{\mathbf{x}}_i$ および $\hat{\mathbf{c}}_i$ はピクセル i の位置および $L^*a^*b^*$ 色情報を表すベクトルである。また σ_s および σ_c は編集マップの一貫性を保つべき位置および色近傍の範囲を制御するパラメータである。式 (1) は凸関数をなすので、各 \hat{e}_i の偏微分が全て 0 になる点でエネルギー関数が最小値を取る。よって、式 (1) を最小化する \hat{e}_i はベクトル形式で次のように得られる。

$$\hat{\mathbf{e}} = \frac{1}{2\lambda} (\hat{D} - \hat{Z})^{-1} \hat{Z} \hat{W} \hat{\mathbf{g}} \quad (2)$$

ここで \hat{Z} は \hat{z}_{ij} を成分に持つ行列であり、 \hat{D}, \hat{W} は対角成分にそれぞれ $\hat{d}_i = \sum_{j=1}^n (\hat{z}_{ij} + \hat{z}_{ij} \hat{w}_j (2\lambda))$, \hat{w}_i を持つ対角行列である。また、 $\hat{\mathbf{g}}$ はピクセル i に入力された編集パラメータ g_i を成分に持つベクトルを表す。式 (2) は大規模密行列 \hat{Z} を含むため、一般の計算機によってこれを計算することは困難である。そこで \hat{Z} が低ランクかつ半正定値であるという特徴を用いた近似法である Nyström 法を用いて近似を行う [1]。

3.2 継承行列による編集マップの移送

アンカーフレームにおいて計算された編集マップは継承行列によって、隣接するフレームから順々に伝搬されていく。継承行列は、隣接する二枚のフレームから計算される行列であり、2枚のフレームが持つピクセルの色を縦に並べたベクトルを $\mathbf{I}^t, \mathbf{I}^{t+1}$ のように置くと、

$$\mathbf{I}^{t+1} = M^{t+1/t} \mathbf{I}^t \quad (3)$$

のような関係を持つ。式 (3) において、 $M^{t+1/t} \in \mathbb{R}^{n \times n}$ が継承行列であり、右上の添え字は、フレーム t からフレーム $t+1$ への継承を表す行列であることを示す。

提案手法では、各フレームに与えられるべき編集マップが式 (3) のような関係を持つと仮定する。すなわち、フレーム t における編集マップを \mathbf{e}^t のように表すことにすれ

ば、次のような関係式が成り立つことを仮定している。

$$\mathbf{e}^{t+1} = M^{t+1/t} \mathbf{e}^t \quad (4)$$

ここで解くべき問題は、このような線形の関係式を導くような継承行列 $M^{t+1/t}$ をどのように導くかということである。今、あるエネルギー関数の最適解として、式 (4) が与えられると考えると、そのエネルギー関数はある行列に関する二次形式をしていると考えられる。そのようなエネルギー関数を定義するために、次のような仮定を置く。

- (i) 隣接する二枚のフレームにおいて同じ位置にあるピクセルが、類似した色を持つ場合には、その二つのピクセルがもつ編集パラメータも類似している。
 - (ii) ある一枚のフレームに注目すると、そのフレーム内で距離が近く、色が類似している場合には、その二つのピクセルが持つ編集パラメータも類似している。
- この仮定をもとに次のような二つの重みづけ関数を定義する。

$$w_i^{t+1/t} = \exp(-\|\mathbf{c}_i^{t+1} - \mathbf{c}_i^t\|_2^2 / (2\sigma_{ct}^2)) \quad (5)$$

$$z_{ij}^{t+1} = \exp(-\|\mathbf{x}_i^{t+1} - \mathbf{x}_j^{t+1}\|_2^2 / (2\sigma_s^2)) \cdot \exp(-\|\mathbf{c}_i^{t+1} - \mathbf{c}_j^{t+1}\|_2^2 / (2\sigma_c^2)) \quad (6)$$

式 (5) および (6) において、 \mathbf{c}_i^t はフレーム t の位置 i に存在するピクセルの $L^*a^*b^*$ 色情報を表し、 \mathbf{x}_i^t は同様のピクセルの位置を表す。またこれらの式において用いられている三種類のパラメータ σ_{ct}, σ_s および σ_c は各成分の違いをどの程度許容するかを制御している。実験において用いられているパラメータは表 1 に示した。

これらの重み付け関数を用いて、次のようなエネルギー関数を定義する。

$$E = \sum_{i=1}^n w_i^{t+1/t} (e_i^{t+1} - e_i^t)^2 + \lambda \sum_{i=1}^n \sum_{j=1}^n z_{ij}^{t+1} (e_i^{t+1} - e_j^{t+1})^2 \quad (7)$$

このエネルギー関数も式 (1) 同様に凸関数をなすため、これを最小化するような解は一つに決まる。式 (7) を各 $e_i^{t+1}, i = 1, \dots, n$ で偏微分することで得られる n 本の方程式が 0 になるとすると、

$$\left(\frac{w_i^{t+1/t}}{2\lambda} + \sum_{j=1}^n z_{ij}^{t+1} \right) e_i^{t+1} - \sum_{j=1}^n z_{ij}^{t+1} e_j^{t+1} = \frac{1}{2\lambda} w_i^{t+1/t} e_i^t$$

という関係式が得られる。これを改めてベクトルと行列を用いて書き直すと次のようになる。

$$(D^{t+1/t} - Z^{t+1}) \mathbf{e}^{t+1} = \frac{1}{2\lambda} W^{t+1/t} \mathbf{e}^t$$

$$\mathbf{e}^{t+1} = \frac{1}{2\lambda} (D^{t+1/t} - Z^{t+1})^{-1} W^{t+1/t} \mathbf{e}^t \quad (8)$$

ただし $D^{t+1/t}$ は $n \times n$ の対角行列で、その対角要素が

$$d_i^{t+1/t} = \frac{w_i^{t+1/t}}{2\lambda} + \sum_{j=1}^n z_{ij}^{t+1} \quad (9)$$

となっている。また $W^{t+1/t}$ も同様に $n \times n$ の対角行列で、その対角要素は $w_i^{t+1/t}$ である。この関係式は式 (4) と同じ形をしており、目的の継承行列が求められている。この継承行列も計算のためには大規模密行列 Z^{t+1} の積を計算する必要があり、式 (2) 同様に Nyström 法を用いた近似が必要である。次節では、近似法の詳細について説明を行う。

3.3 継承行列の近似

継承行列を含め、式 (2) なども計算には大規模密行列 $Z^{t+1/t}$ あるいは \hat{Z} の計算を行う必要がある。ところが、これらの行列は画像のピクセル数 n に対して $n \times n$ の大きさであり、一般的な計算機で計算を行うことは、行列を保持するためのメモリと計算時間の両面から困難である。以下では、簡単のために、行列の添え字を省略する。

Nyström 法の適用方法。 Nyström 法を用いると行列が低ランクである場合、少ない数の固有値と固有ベクトルだけを用いて効率的に行列を近似することができる。この固有値および固有ベクトルを近似的に求めるために Nyström 法では元の行列の列ないし行をサンプリングする [13]。本手法は 2 枚の隣接するフレームごとに Nyström 法による近似が必要であるため、従来の列をサンプリングする Nyström 法では精度が不十分であった。そのため、本研究では Zhang らが提案した k -means 法に基づく Nyström 法を用いる。

まず、入力されたフレームをピクセルの $L^*a^*b^*$ 色情報と位置の情報に基づき k -means 法でクラスタリングする。本来は、行列 Z のランクを用いて k 個のクラスタに分割するのが理想的であるが、実際にこの k を求めるには、ある程度の計算量が必要であること、無視しても差し支えない小さな固有値により k が必要以上に大きくなってしまふことを防ぐためサンプル数 m を設定する。入力データを m 個のクラスタに分解したとすると、それぞれのクラスタについてクラスタ中心 $(\bar{\mathbf{c}}_j^T, \bar{\mathbf{x}}_j^T)^T, j = 1, 2, \dots, m$ が得られる。これを用いて、次の行列 $U \in \mathbb{R}^{n \times m}$ および $A \in \mathbb{R}^{m \times m}$ を計算する。

$$\begin{aligned} U_{ij} &= \exp(-\|\mathbf{x}_i^{t+1} - \bar{\mathbf{x}}_j^{t+1}\|_2^2 / (2\sigma_s^2)) \\ &\quad \cdot \exp(-\|\mathbf{c}_i^{t+1} - \bar{\mathbf{c}}_j^{t+1}\|_2^2 / (2\sigma_c^2)) \\ A_{jl} &= \exp(-\|\bar{\mathbf{x}}_j^{t+1} - \bar{\mathbf{x}}_l^{t+1}\|_2^2 / (2\sigma_s^2)) \\ &\quad \cdot \exp(-\|\bar{\mathbf{c}}_j^{t+1} - \bar{\mathbf{c}}_l^{t+1}\|_2^2 / (2\sigma_c^2)) \end{aligned}$$

ただし $1 \leq i \leq n, 1 \leq j, l \leq k$

この行列 U および A を用いると、Nyström 法により $Z \approx UA^{-1}U^T$ という近似が得られる。

継承行列の近似に必要な残りの計算は $(D - Z)^{-1} \approx (D - UA^{-1}U^T)^{-1}$ であるが、これは Woodbury の公式あるいは逆行列の補題と呼ばれる定理を用いることで次のよ

うに計算が可能である。

$$(D - UA^{-1}U^T)^{-1} = D^{-1} - D^{-1}U(-A + U^T D^{-1}U)^{-1}U^T D^{-1}$$

この式は、小さな行列同士の積のみから成り立っており、実際に計算が可能である。以上の計算方法を用いることで、継承行列は以下のように近似される。

$$\begin{aligned} M^{t+1/t} &\approx \frac{1}{2\lambda} (D^{-1} - D^{-1}U(-A + U^T D^{-1}U)^{-1}U^T D^{-1}) W \\ D &= \left(\frac{1}{2\lambda} W + Z \right) \mathbf{1}_n \approx \left(\frac{1}{2\lambda} W + UA^{-1}U^T \right) \mathbf{1}_n \end{aligned}$$

この近似した継承行列をアンカーフレームにおいて計算された編集マップに順々に適用していくことで、最終的な動画編集の結果を得る。

3.4 動画編集の結果

提案手法による動画編集の結果を図 2, 3 および 4 に示す。それぞれの結果から、継承行列を用いた動画編集法の特徴について説明を行う。

まず、図 2 に注目すると、入力を行ったフレームに写りこんでいなかった物体についても同系統の色調を有するものを自動的に編集できていることがわかる。これはエネルギー関数である式 (7) の第二項においてフレーム内での編集の一貫性を考慮しているためで、新たな物体が入り込んでくるたびにユーザ入力を追加しなくても良いというのは動画編集において利点となるといってもよい。

続いて図 3 にはテキスト強調の結果を示した。このテキスト強調は定義域変換 (domain transform) によるエッジ保存型平滑化フィルタ [6] を用いて各フレームをマルチスケール分解 (multiscale decomposition) した後、どのスケールの詳細をどの程度、強調・抑制するかを決定するパラメータを継承行列により伝搬している。この結果からもわかる通り、本手法はどのような編集を行うかという目的に依存せず、その編集が独立したいくつかのパラメータにより行われている場合にはいかなる編集法にも適用が可能である。

最後に図 4 には複数のアンカーフレームにユーザ入力を行った場合の編集結果を示した。通常、複数のフレームにユーザ入力を行う目的は、フレームの進行に伴い蓄積したエラーが原因で起こる不適切な領域分割を防ぐためであることが多い。本手法でもそのような目的でユーザ入力を追加することは可能であるが、実験を行ったいくつかの動画について、編集パラメータの伝搬は適切かつ安定に行われており、その必要性は薄いと考えられる。その一方で複数のアンカーフレームに異なる編集を施すことで、より自由度の高い編集を行うことが可能であり、図 4 には元の動画において黄色であるモミジの葉を赤色に色づかせるという編集を行っている。このように、編集パラメータを安定的に伝搬できるようになったことで、従来は行うことの難しかった複数フレームに対するユーザ入力の可能性を示すこ

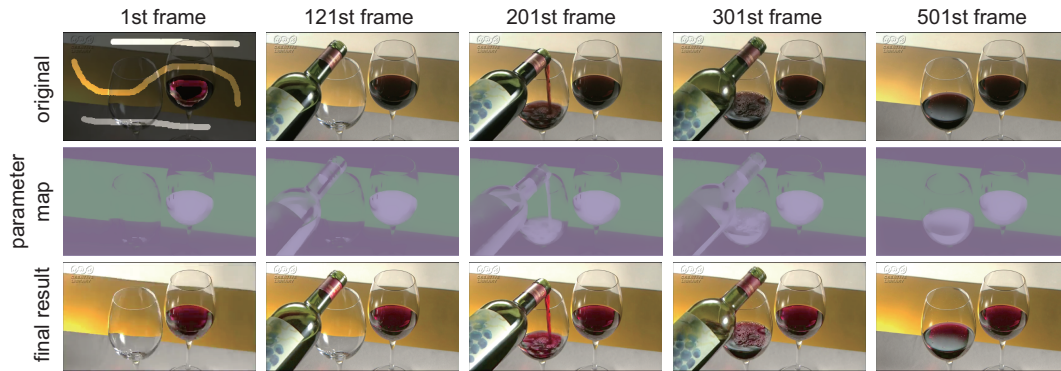


図 2 継承行列による動画の色調補正結果. この結果は最初のフレームにのみユーザ入力を行って作成したものである. 行ったユーザ入力は上段の一番左に示した. この動画では最初のフレームに写っている片方のワイングラスにしか入力を入れていないが, のちに現れるワインボトルから注がれるワインについても適切に色が編集されている.

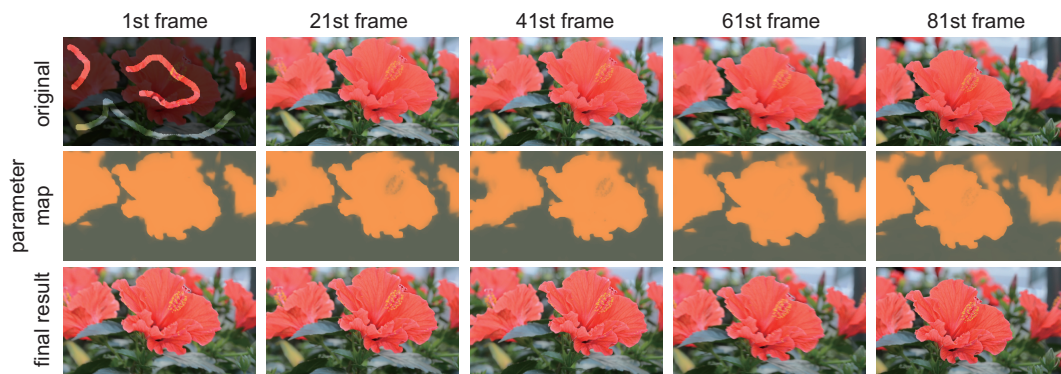


図 3 動画に対するテクスチャ強調の結果. この結果は継承行列により伝搬したパラメータに基づいて, 定義域変換によるエッジ保存型平滑化フィルタ [6] を用いたマルチスケール分解でテクスチャ強調を行っている. この動画では花の部分のテクスチャが強調されている一方で背景や葉の部分にはぼかしをかけている.

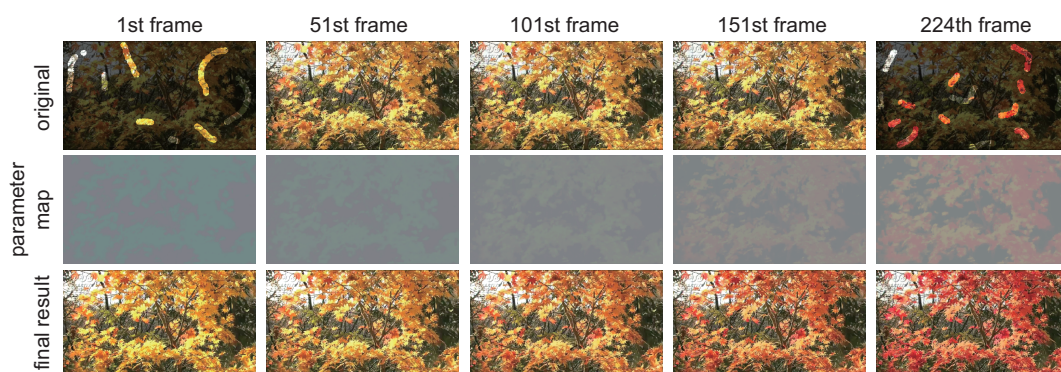


図 4 2つのアンカーフレームを用いた色調変換の結果. 元の動画では黄色のモミジの葉が風で揺られているが, 最初と最後のフレームに異なる色調変化を与えるユーザ入力を行うことで徐々に葉の色が変化するように編集を行った. この結果は, 2つのアンカーフレームから得られるパラメータを線形補間することにより作成しているが, 補間の方法は線形に限らず, 例えばシグモイドカーネルのようなものを用いて補間を行ってもよい.

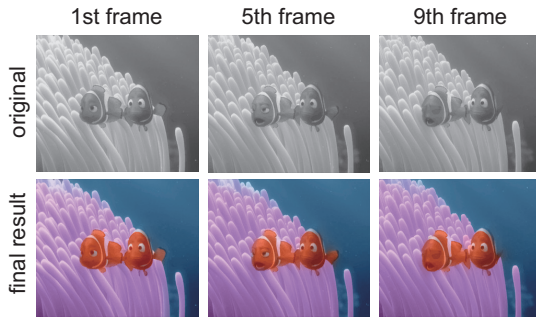


図 5 グレースケール動画に対する着色処理. この結果はアンカーフレームに対してグラフ探索によるグレースケール画像の着色処理 [14] を行い, その結果を継承行列により伝搬したものである.

とができた.

4. 提案手法の応用

本研究が提案する継承行列を用いた動画編集法は処理の過程を少し変更するだけで, 様々な応用が可能である. 本節では, その応用例として, グレースケール動画に対する着色処理, 動画の色調安定化, および静止画から動画への色転写の手法を紹介する.

4.1 グレースケール動画の着色処理

グレースケールの静止画に色をつけるという処理は, 従来から多くの研究が存在しており, 大きく分けると, ユーザ入力をもとにした着色処理 [9, 14] と参照画像を用いた処理 [7] に分けられる. ここでは, 前者のユーザ入力をもとにした手法を継承行列を用いて動画に拡張する.

着色処理に対し, 式 (7) をそのまま用いようとすると, 第二項のフレーム内の一貫性を保つ項が強く働きすぎてしまい, 全体的にぼやけた着色結果となってしまふ. これは, グレースケールの画像ないし動画において, 全てのピクセルが輝度値の一次元情報のみを持っており, なおかつ全体でその輝度値がそれほど大きくは変わらないということに原因があると考えられる. そこで, 本手法を着色処理に用いる場合には, 式 (7) の第二項をピクセルの近傍のみとの一貫性を保つように変更する. 具体的には以下のような式となる.

$$E = \sum_{i=1}^n w_i^{t+1/t} (e_i^{t+1} - e_i^t)^2 + \lambda \sum_{i=1}^n \sum_{j \in \mathcal{N}_4(i)} z_{ij}^{t+1} (e_i^{t+1} - e_j^{t+1})^2$$

ただし, この式において $\mathcal{N}_4(i)$ はピクセル i の四近傍を指す. この式を用いて得られた動画の着色結果を図 5 に示す. この結果はアンカーフレームに対して, グラフ探索による静止画像の着色処理 [14] において行い, その結果を継承行列で伝搬している.

4.2 色調安定化処理

動画の色調安定化処理とは, 安価なデジタルカメラや携

帯電話で動画を撮影した際に起こる色度の揺れを除去する処理である. 近年, Farbman と Lischinski により提案されたこの処理は, 基準となるアンカーフレームから他のフレームの各画素がどの程度色度のずれを起こしているかというのを計算することで実現される [4]. 従来法においては, 各フレームごとに信頼度マップやアピランスマップと呼ばれる複数のパラメータマップを計算する必要があったが, 継承行列を用いることで, これらの処理を単純化することが可能である.

あるフレーム t において, 各ピクセルがアンカーフレームからどの程度の色度のずれを持つかという情報を e_i^t と表すことにする. フレームの進行に伴い, 色度のずれは蓄積されていくと考えられるので, その増加分をフレーム間の色度の差により $L_i^{t+1} - L_i^t$ のように表す. すると, フレーム $t+1$ における色度のずれは次のようにして与えられる.

$$e^{t+1} = M^{t+1/t} (e^t - \mathbf{L}^t + \mathbf{L}^{t+1})$$

ただし \mathbf{L}^t はフレーム t における各ピクセルの色度 L_i^t を成分に持つベクトルで $\mathbf{L}^t = (L_1^t, L_2^t, \dots, L_n^t)^T$ を表す. このようにして得られた動画の色調安定化の結果を図 6 に示す.

4.3 画像から動画への色転写

色転写 (color transfer) とは, 目的とする静止画の色調を参照する静止画の色調に合わせるという手法である [12]. 色転写の基本的な考え方は, 画像をヒストグラムやガウス混合モデルなどの確率モデルにし, 目的画像の確率モデルを参照画像の確率モデルに近づけることで, 目的画像の色調を変化させるというものである.

本手法による静止画から動画への色転写は, まずユーザ入力により領域間の対応付けを行い, その領域間で Reinhard のガウス分布に基づいた色転写のパラメータを計算する. Reinhard の手法は $L^*a^*b^*$ の各色成分が目的画像と参照画像の両方でガウス分布をしていると仮定する. 今, 目的画像の k 番目の色成分の平均を μ_k^{obj} , 標準偏差を σ_k^{obj} , 参照画像の k 番目の色成分の平均を μ_k^{ref} , 標準偏差を σ_k^{ref} と表すことにする. 目的画像のピクセル i における元の色を $c_{k,i}^{obj}$ とすると, 色転写後の目的画像の色 $\hat{c}_{k,i}^{obj}$ は次のように表される.

$$\hat{c}_{k,i}^{obj} = \frac{\sigma_k^{ref}}{\sigma_k^{obj}} (c_{k,i}^{obj} - \mu_k^{obj}) + \mu_k^{ref} \quad (10)$$

式 (10) は $a_k = \frac{\sigma_k^{ref}}{\sigma_k^{obj}}$, $b_k = \mu_k^{ref} - \frac{\sigma_k^{ref}}{\sigma_k^{obj}} \mu_k^{obj}$ とおくことで $\hat{c}_{k,i} = a_k c_{k,i}^{obj} + b_k$ と表せる. そこで本研究では $L^*a^*b^*$ 色空間で各成分について a_k および b_k を計算し, 計 6 次元の情報を継承行列により伝搬した. これにより得られた色転写の結果を図 7 に示す.

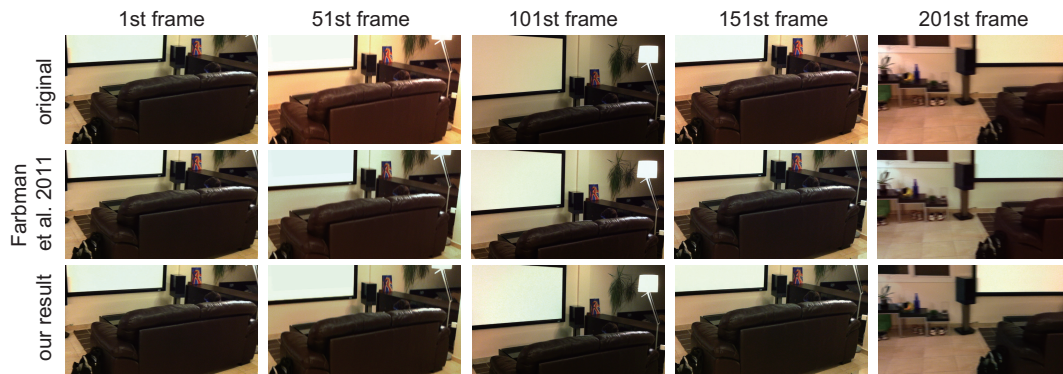


図 6 動画の色調安定化結果. この結果では入力動画において見られる色度の揺れが, 出力動画において除去されていることがわかる. また従来手法 [4] と比べるとわずかな違いではあるがソファ周辺の色がより安定していることがわかる.

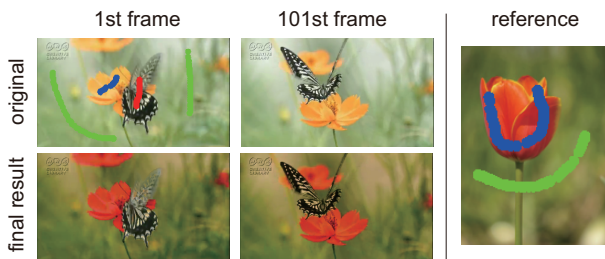


図 7 継承行列を用いた静止画から動画への色転写の結果. 入力動画の 1 フレーム目および参照画像にユーザ入力が行われている. 青および緑の入力は領域間の対応関係をとっており, 入力動画の方にのみ存在する赤の入力は蝶の部分の色調が変化しないようにするためのものである.

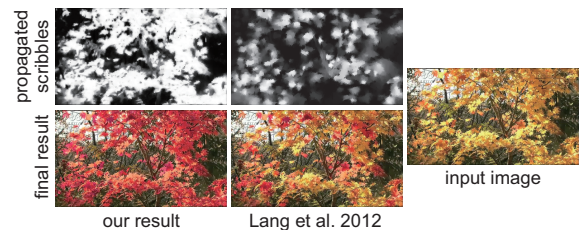


図 8 Lang らによる近似的なオプティカル・フローを用いた編集マップの伝搬法 [8] との比較. この画像は同じ編集マップを 15 フレーム分伝搬した結果であり, 従来法が部分的にしか適切な伝搬を行っていない一方で, 提案手法は適切かつ安定に伝搬を行っている.

5. 考察

5.1 継承行列の性質

継承行列 $M^{t+1/t}$ は添え字などを省略すると, 対称半正定値行列 $Z \in \mathbb{R}^{n \times n}$ と二つの対角行列 $D, W \in \mathbb{R}^{n \times n}$, および実数定数 $\lambda \in \mathbb{R}$ を用いて

$$M^{t+1/t} = \frac{1}{2\lambda}(D - Z)^{-1}W$$

の形で表せる. この行列は次のような特性を持つ.

- (i) 行列 $D - Z$ が正則であり, かつ W が 0 かつ非負の対角成分を持つため $M^{t+1/t}$ も正則であり, ゆえに逆行列 $(M^{t+1/t})^{-1}$ が存在する.
- (ii) 継承行列 $M^{t+1/t}$ に逆行列が存在すれば式 (4) から, それはフレーム $t+1$ からフレーム t への逆方向のパラメータの継承を表す. すなわち $(M^{t+1/t})^{-1} = M^{t/t+1}$ である.
- (iii) フレーム t からフレーム $t+k$ までの間の継承行列 $M^{t+1/t}, \dots, M^{t+k-1/t+k}$ の積はフレーム t からフレーム $t+k$ へのパラメータの継承を表す. すなわち $\prod_{i=t}^{t+k-1} M^{i+1/i} = M^{t+k-1/t+k} \dots M^{t+1/t} = M^{t/t+k}$ である.

これらの性質のうち, (i) および (ii) の性質はオプティカ

ル・フローなどの位置の対応関係を考えた場合には存在しない性質である. 位置の対応関係を行列として表現する場合を考えると, フレーム間でピクセルとピクセル同士の一対一の対応関係が与えられている場合のみその行列は逆行列を持つ, すなわちフレーム $t+1$ のあるピクセル i がフレーム t の二つ以上のピクセルと対応関係を持つ場合には, その行列表現が逆行列を持たない. これは位置の対応関係を取る場合においてしばしば起こりうることであり, その点で継承行列は性質が優れた行列であるといつてよい.

また, (iii) の性質は例えばあるフレームに対して行ったユーザ入力から 100 フレーム後の結果をいきなり求めたい場合などに役に立つ. 実際には, $M^{t/t+k}$ という行列は大規模な行列であり, あらかじめ計算しておくことはできない. そのため, フレームの順序にしたがって結果を求めていくことになるのだが, もし継承行列をメモリ上に保存しておくことができれば, 継承行列をより有用に扱うことができると考えられる.

5.2 編集伝搬の精度とパラメータ

まずフレーム内での編集伝搬の精度であるが, 従来法 [1] などでも報告されている通り, あまりに細かなユーザ入力を適切に伝搬することは難しい. 細かなユーザ入力を区別するためには一貫性を保つべきピクセルの近傍範囲を制御

表 1 本手法の実験において用いたパラメータ

	σ_s	σ_c	σ_{ct}	m	s
図 2, 3, 4, 7	50	1.2	3.0	50	5
図 5	—	0.5	0.5	—	—
図 6	120	1.2	5.0	60	5

するパラメータ σ_s を小さな値にすれば良いのだが、この σ_s が 0 に近づくと類似度行列 Z の低ランクであるという性質が失われてしまい、Nyström 法による近似の精度が低下する。このため、細かな編集を伝搬することは難しい。

一方で、時間方向の伝搬の精度は従来法と比べて著しい改善が行えている。近似的なオブティカル・フローにより領域ラベルの伝搬を行う従来法 [8] との比較を図 8 に示した。従来法は細かなテクスチャに対応できず、またフレームの進行に伴うエラー蓄積も大きい。本手法では各結果画像に示した通り一般的な動画で 5 秒程度にあたる 150 フレームを超えてもおおむね良好に編集の伝搬が行えている。

ただし、従来法がフレーム間の伝搬を 1280×720 ピクセルのフレームに対して 0.7 秒程度で行えるのに対し、本手法は 360×240 ピクセルのフレームに対し同程度の時間を要する。またメモリの使用量についても従来法が動画のフレーム数 T 、幅 W 、高さ H に対して、 $O(\max(T, W, H))$ であるのに対し、本手法は Nyström 法におけるサンプル数 m を用いて $O(WHm)$ であり、より多くのメモリを必要とする。なお本手法の実装は MATLAB および C++ を用いて行っており、実験は 3.6GHz の CPU 上で行った。実験で用いたパラメータは表 1 に示した。

6. まとめ

本報告では、フレーム間の連続性を保持した動画編集法の紹介を行った。本研究では従来の動画を扱う処理において一般的であるとされてきたフレーム間の位置の対応関係に頼らず、継承行列を用いたフレーム間の線形結合性によってこれを実現した。継承行列による動画編集は従来法と比べ安定的に編集の伝搬を行うことが可能で、本手法により動画の編集がより少ない手間で出来るようになることが期待される。また継承行列の適応範囲は、色調補正やテクスチャ強調といった単純な編集に限らない。グレースケール動画の着色や色調の安定化、静止画像から動画への色転写など、様々な用途への応用が可能である。

本手法は HD など高解像度の動画に対して直接適用するには計算時間やメモリの使用量などいくつかの問題を抱えているが、フレームをダウンサンプルしたものに対して計算した編集マップをアップサンプルするなどの処理により、高解像度の動画にも応用可能である。ただし、この場合には細かな部分でアップサンプルによるノイズなどが生じる可能性があることは言うまでもない。

このような問題を考慮してもなお、本手法の安定性や適

応範囲の広さは有用であると考えられ、本研究により動画の編集がより少ない手間で簡単に行えるようになる足がかりとなれば幸いである。

謝辞

本報告において使用されている動画のサンプルの多くは NHK クリエイティブ・ライブラリー (<http://www.nhk.or.jp/creative/>) にて公開されているものを使用しており、これらのサンプルを公開している NHK に感謝の意を表す。色調安定化の項における従来法の比較データは Zeev Farbman 氏から提供していただき、これにより従来法との比較が可能になった。この場を借りて彼に感謝の意を表す。また本研究は日本学術振興会基盤研究 B (22300030) より支援を受けて行われた。関係者の皆様に感謝の意を表す。

参考文献

- [1] An, X. and Pellacini, F.: AppProp: all-pairs appearance-space edit propagation, *ACM Trans. Graph.*, Vol. 27, No. 3, pp. 40:1–40:9 (2008).
- [2] An, X. and Pellacini, F.: User-Controllable Color Transfer, *Computer Graphics Forum*, Vol. 29, No. 2, pp. 263–271 (2010).
- [3] Bai, X., Wang, J., Simons, D. and Sapiro, G.: Video SnapCut: robust video object cutout using localized classifiers, *ACM Trans. on Graphics*, Vol. 28, pp. 70:1–70:11 (2009).
- [4] Farbman, Z. and Lischinski, D.: Tonal stabilization of video, *ACM Trans. Graph.*, Vol. 30, No. 4, pp. 89:1–89:10 (2011).
- [5] Fattal, R., Lischinski, D. and Werman, M.: Gradient Domain High Dynamic Range Compression, *ACM Trans. Graph.*, Vol. 21, No. 3, pp. 249–256 (2002).
- [6] Gastal, E. S. L. and Oliveira, M. M.: Domain transform for edge-aware image and video processing, *ACM Trans. Graph.*, Vol. 30, No. 4, pp. 69:1–69:12 (2011).
- [7] Irony, R., Cohen-Or, D. and Lischinski, D.: Colorization by example, *Proc. of Eurographics Symposium on Rendering*, pp. 201–210 (2005).
- [8] Lang, M., Wang, O., Aydin, T., Smolic, A. and Gross, M.: Practical temporal consistency for image-based graphics applications, *ACM Trans. Graph.*, Vol. 31, No. 4, pp. 34:1–34:8 (2012).
- [9] Levin, A., Lischinski, D. and Weiss, Y.: Colorization using optimization, *ACM Trans. Graph.*, Vol. 23, No. 3, pp. 689–694 (2004).
- [10] Lischinski, D., Farbman, Z., Uyttendaele, M. and Szeliski, R.: Interactive local adjustment of tonal values, *ACM Trans. Graph.*, Vol. 25, No. 3, pp. 646–653 (2006).
- [11] Liu, F. and Gleicher, M.: Learning color and locality cues for moving object detection and segmentation, *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 320–327 (2009).
- [12] Reinhard, E., Adhikhmin, M., Gooch, B. and Shirley, P.: Color transfer between images, *Computer Graphics and Applications, IEEE*, Vol. 21, No. 5, pp. 34–41 (2001).
- [13] Williams, C. and Seeger, M.: Using the Nyström Method to Speed Up Kernel Machines, *Advances in Neural Information Processing Systems 13*, pp. 682–688 (2001).
- [14] Yatziv, L. and Sapiro, G.: Fast image and video colorization using chrominance blending, *IEEE Trans. on Image Processing*, Vol. 15, No. 5, pp. 1120–1129 (2006).

- [15] Zimmer, H., Bruhn, A. and Weickert, J.: Optic Flow in Harmony, *International Journal of Computer Vision*, Vol. 93, pp. 368–388 (2011).