

# 手描きアニメ彩色のための 形状特徴量に基づく領域マッチング

金森 由博<sup>a)</sup>

**概要:** 手描きアニメの彩色作業は、線画中の各閉領域を手作業で指定して塗りつぶすという方法で行われている。これを効率化するために、本研究では連続したフレームの線画において同じ色で塗られるべき閉領域の対応付けを行う。まず、各閉領域の形状から得られる特徴量を計算し、この特徴量から2つの閉領域の対応付けのコストを計算する。このコストに基づいて、対応付けのコストの総和が最小になるような組み合わせを計算する。本研究では特徴量を計算するための形状記述子として、各閉領域から計算される楕円、フーリエ記述子と Shape Context の3つを検討し、それぞれを用いた対応付けの精度について検討する。

## 1. はじめに

2次元手描きアニメはその誕生のときから人々を魅了してきた。しかしながら、制作支援ソフトウェアが現れた今なお、手描きアニメの制作は多大な時間を要する労働集約的な作業である。その制作には次の工程が含まれる。すなわち、紙に線画を手描きし、各フレームごとにスキャンする。そしてスキャンした線画をクリーンアップしたのち、線画の閉領域をひとつずつ手作業で指定し、塗っていく。この塗りの作業は他の工程に比べ単純であり、自動化が望まれる。

着彩工程を自動化するには、連続するフレーム間で閉領域の対応付けを行い、対応する領域に同じ色を伝播させればよい。本研究でもこの方法を採用する。既存手法の多くは、閉領域の輪郭線上の特徴点から得られる特徴量を用いて閉領域の対応付けを行なっている [2], [4]。ここでいう特徴点とは、線が交差している点や曲率が大きく変化している点などを指す。しかし輪郭線上の特徴点は、アニメの動きに伴う形状変形の影響を受けやすく、結果として対応付けの精度が下がる。本研究では、輪郭線の形状変形の影響を受けにくい、次の3つの形状記述子について比較検討する。

- (1) 閉領域の形状を粗く近似する楕円 [3]
- (2) 閉領域の形状の低周波成分を表すフーリエ記述子 [11]

- (3) 閉領域の形状を極座標表示し、対数距離と偏角の2次元ヒストグラムで表現する Shape Context [1]

上記のうち文献 [3] の手法は、もともと手描きアニメにおける閉領域の対応付けのために提案されたものである。一方、フーリエ記述子 [11] と Shape Context [1] については、ロゴマークや道路標識などの入力形状に対し類似した形状を取得する、という目的で開発されたため、平行移動、拡大縮小、回転などの座標変換に対して不変となるような工夫が施されている。しかし本研究の目的では、対応付けのために座標変換の影響も加味する必要があると考えられるため、適宜修正を加える。これらの形状記述子から2つの閉領域の対応付けのコストを導出し、連続する2フレーム間での閉領域の最適な組み合わせを2部グラフマッチングにより求める。

本稿では、実際の手描きアニメの画像データを用い、これらの精度について比較実験を行った結果を報告する。

## 2. 関連研究

複数のフレーム間で閉領域を一度に塗る単純な方法として、「串刺しツール」と呼ばれるものがある。この機能は、複数のフレームを重ね、ユーザが指定した座標を含む閉領域をすべて同じ色で塗る、というものである。この機能は塗ろうとする複数の閉領域が重なっていなければならず、指定通り一度に塗ることに失敗しやすいため、制作の現場ではあまり使われていない。

既存手法では閉領域ごとの対応付けに、輪郭線の形状を用いている [2], [4]。輪郭線の形状記述子として、輪郭線の変化を文字列として符号化したものや、*dominant points* と呼ばれる点列が用いられる。しかし、これらは輪郭線の

<sup>1</sup> 筑波大学  
University of Tsukuba, Tennoudai 1-1-1, Tsukuba, Ibaraki,  
305-0006, Japan

<sup>†1</sup> 現在、筑波大学  
Presently with University of Tsukuba

<sup>a)</sup> kanamori@cs.tsukuba.ac.jp

形状が大きく変化する場合には有効でない。

閉領域の対応付けのために、線画に描かれた物体の骨格構造を利用する手法もある [6]。しかし骨格構造は関節を持つキャラクターにしか適用できない。

Sýkora ら [7] は、2つのフレーム間で画像の位置合わせを行うことにより、手描きアニメに彩色やテクスチャを付加することのできる手法を提案した。彩色には彼らの提案した *LazyBrush* [9] を用いる。しかし、彼らの手法では剛体的形状一致法 [8] を用いているため、2フレーム間の位置合わせごとに数秒の時間がかかり、物体が非剛体的に変形する場合には対応しづらい。

一般に、対応付けに用いる特徴量についてはコンピュータビジョンの分野において盛んに研究されている。ただし、本研究で扱う画像は線画であり、自然画像のような濃淡情報は含まないため、適用できるのは線画から得られる形状特徴量に限られると考えられる。形状特徴量について詳しくは文献 [10]などを参照されたい。本研究では、昨年提案された楕円に基づく形状記述子 [3] と、代表的な形状記述子である、フーリエ記述子 [11] および Shape Context [1] を用いて形状特徴量を計算し、対応付けの精度について比較検討を行う。

### 3. 対応付けの基本的な流れ

本研究では3つの形状記述子について比較を行うが、基本的な処理の流れは同じである。入力として、彩色支援のためには線画像を入力とすべきであるが、線画だけでは正確な塗り分けが行えたかどうか判断できない。そこで本研究では、彩色済みの画像を入力とし、色ごとに領域分割を行って、対応付けの正解データと各閉領域を同時に取得する。なお、領域分割の際には領域の輪郭を表す線（主に黒で描かれる）は除外する。また、閉領域はそれぞれ一色で塗られているものと仮定する。領域分割が済んだら、連続する2フレーム間の閉領域について、どの閉領域とどの閉領域とが対応するかを計算する。全フレームの対応付けが終わると、各閉領域はフレーム間で数珠繋ぎに対応付けられる。ひとつの閉領域に色を割り当てると、その数珠繋ぎの対応関係に沿って、その色が連続するフレームに伝播される。

対応付けについてより詳しく説明する。連続する2つのフレーム  $f$  および  $f+1$  に、それぞれ  $N_f$  個および  $N_{f+1}$  個の閉領域があるとする。対応付けのために、コスト行列  $A = \{a_{ij}\}$  を計算する。このコスト行列の各要素  $a_{ij}$  は、フレーム  $f$  内の閉領域  $i$  とフレーム  $f+1$  内の閉領域  $j$  とを対応付けるためのコストである（ただし  $i = 1, 2, \dots, N_f$  かつ  $j = 1, 2, \dots, N_{f+1}$ ）。このようにして得られた  $A$  に対し、ハンガリー法 [5] によって2部グラフマッチングを解く。2フレームの閉領域の数が一致しない場合、すなわち  $N_f \neq N_{f+1}$  であるとき、いくつかの閉領域は対応付けされ

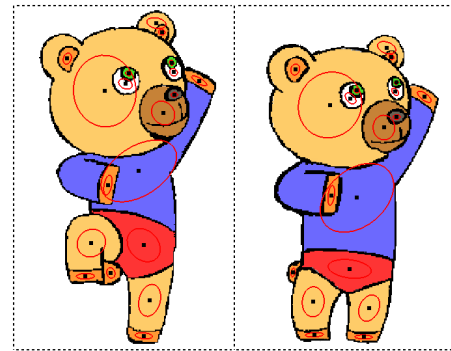


図1 楕円形状記述子。文献 [3] の手法では、各閉領域を近似する楕円（赤）に基づいて、連続する2フレーム間で閉領域ごとの対応付けを行う。

ないままになる。これは、どちらかのフレームにおいて閉領域が新たに出現したか消失したかのどちらかであると解釈できる。

本研究では、上記のコスト行列  $A = \{a_{ij}\}$  の各要素  $a_{ij}$  の計算において、形状記述子ごとに3通りの方法を比較検討する。以下、それぞれの形状記述子について説明する。

### 4. 楕円形状記述子

文献 [3] で提案された手法では、各閉領域を楕円で近似する（図1）。閉領域  $i$  を近似する楕円  $i$  は、閉領域  $i$  内の各画素の位置の重心  $\mathbf{t}_i$  および共分散行列  $C_i$  から算出される。ここで、共分散行列  $C_i$  の最大固有値を  $\lambda_i^{max}$ 、最小固有値を  $\lambda_i^{min}$  とし、それらに対応する固有ベクトルをそれぞれ  $\mathbf{e}_i^{max}$  および  $\mathbf{e}_i^{min}$  とする。すると楕円  $i$  の長軸、短軸の長さはそれぞれ  $\sqrt{\lambda_i^{max}} \mathbf{e}_i^{max}$  および  $\sqrt{\lambda_i^{min}} \mathbf{e}_i^{min}$  で与えられる。

対応付けコスト  $a_{ij}$  は、2つの閉領域  $i, j$  を近似する楕円の、向き、大きさおよび位置の違いから、次のように導出する。

$$a_{ij} = w_{angle} a_{ij}^{angle} + w_{scale} a_{ij}^{scale} + w_{pos} a_{ij}^{pos}, \quad (1)$$

$$a_{ij}^{angle} = \cos^{-1}(\mathbf{e}_i^{max} \cdot \mathbf{e}_j^{max}), \quad (2)$$

$$a_{ij}^{scale} = \sqrt{\frac{\max\{\lambda_i^{max}, \lambda_j^{max}\}}{\min\{\lambda_i^{max}, \lambda_j^{max}\}}} + \sqrt{\frac{\max\{\lambda_i^{min}, \lambda_j^{min}\}}{\min\{\lambda_i^{min}, \lambda_j^{min}\}}} - 2, \quad (3)$$

$$a_{ij}^{pos} = \|\mathbf{t}_i - \mathbf{t}_j\|^2, \quad (4)$$

ここで  $w_{angle}$ ,  $w_{scale}$  および  $w_{pos}$  は重みである。座標  $\mathbf{t}_i$  および  $\mathbf{t}_j$  は画像サイズで正規化される。 $a_{ij}^{angle}$ ,  $a_{ij}^{scale}$  および  $a_{ij}^{pos}$  の各項は、閉領域  $i, j$  を近似する楕円が、まったく同一の向き、大きさおよび位置となる場合にゼロとなるようになっている。対応付けの精度を向上するため、さらに、閉領域  $i, j$  の隣接領域の情報も対応付けコスト  $a_{ij}$  に追加する。この手法では閉領域  $i, j$  それぞれの隣接領域の重心を求め、隣接領域の重心の移動距離を、対応付けコストに加える。閉領域  $i$  の隣接領域の重心  $\mathbf{n}_i$  は次のように算出される。

$$\mathbf{n}_i = \frac{\sum_{k \in \{O_i, i\}} \alpha_k \beta_k \exp\left(-\frac{|t_k - t_i|^2}{\sigma_i^2}\right) \mathbf{t}_k}{\sum_{k \in \{O_i, i\}} \alpha_k \beta_k \exp\left(-\frac{|t_k - t_i|^2}{\sigma_i^2}\right)}, \quad (5)$$

$$\sigma_i = \sigma_0 \max_{k \in O_i} |\mathbf{t}_k - \mathbf{t}_i|, \quad (6)$$

ここで  $O_i$  は閉領域  $i$  に隣接する閉領域の集合である。 $\alpha_k$  は閉領域  $k$  の面積を表す。もし閉領域  $k$  の面積が大きい場合、重心位置に対して閉領域  $k$  の影響が大きくなる。しかし、閉領域  $k$  が閉領域  $i$  とわずかにしか接触していない場合には、重心位置が閉領域  $k$  の移動に影響を受けすぎないように、閉領域  $k$  の重みを小さくしたい。そこで閉領域  $k$  の接触率  $\beta_k \in [0, 1]$  を重みに乗算する。この接触率とは、閉領域  $i$  の輪郭線の全長のうち、どれだけの割合が閉領域  $k$  と接しているかを示す値である。さらに閉領域  $k$  が背景である場合、背景は面積が非常に大きくなりやすいため、背景の影響をなくすべく、 $\alpha_k = \alpha_i$  および  $\mathbf{t}_k = \mathbf{t}_i$  と計算する。また、閉領域  $k$  の重心位置が閉領域  $i$  の重心から離れている場合に閉領域  $k$  の影響を減らすべく、2つの閉領域の重心間も重みとして考慮する。 $\sigma_0$  は係数である。最後に、次の式で計算される項  $a_{ij}^{neighbor}$  に重み  $w_{neighbor}$  を乗算し、対応付けコスト  $a_{ij}$  に加算する。

$$a_{ij}^{neighbor} = |\mathbf{n}_i - \mathbf{n}_j|^2, \quad (7)$$

$$a_{ij} = w_{angle} a_{ij}^{angle} + w_{scale} a_{ij}^{scale} + w_{pos} a_{ij}^{pos} + w_{neighbor} a_{ij}^{neighbor}. \quad (8)$$

## 5. フーリエ記述子

フーリエ記述子には多くの変種が存在する。文献 [11] ではそれらのうち、

**重心距離:** 形状の輪郭線上の動点と重心の距離を計算

**曲率変化:** 形状の輪郭線の曲率の変化を計算

**累積角度:** 形状の輪郭線と接線のなす角を累積

の3つについて、類似形状を検索する実験を行い、重心距離を用いた場合に最もよい結果が得られたと報告している。そこで本研究でも、重心距離に基づいたフーリエ記述子を計算する。

文献 [11] の、重心距離に基づいたフーリエ記述子の計算方法について説明する。まず形状の輪郭線を、等しい長さの  $N$  本の線分からなる折れ線に変換する。折れ線の頂点  $t$  (ただし  $t = 0, 1, \dots, N-1$ ) について、重心からの距離が  $r_t$  であるとする。このとき、第  $n$  項のフーリエ係数  $u_n$  (ただし  $n = 0, 1, \dots, N-1$ ) は次のように計算される。

$$u_n = \frac{1}{N} \sum_{t=0}^{N-1} r_t \exp\left(\frac{-j2\pi nt}{N}\right), \quad (9)$$

ここで  $j$  は虚数単位である。直流成分  $u_0$  は実部のみからなり、形状の位置のみに依存する。文献 [11] では、平行移動、拡大縮小と回転に対して不変になるよう、次のベクト

ル  $\mathbf{f}$  を形状の特徴ベクトルとする。

$$\mathbf{f} = \left( \frac{|u_2|}{|u_1|}, \frac{|u_3|}{|u_1|}, \dots, \frac{|u_{N-1}|}{|u_1|} \right). \quad (10)$$

形状比較の際には、特徴ベクトル  $\mathbf{f}$  の要素のうち最初から  $K$  個 ( $0 \leq K \leq N$ ) を選び、特徴ベクトル間のユークリッド距離を計算することで類似度とする。

### 本研究での計算方法

本研究では、各閉領域の相対位置も考慮したいため、2つの閉領域の重心間の距離も計算する。また、式 (10) では拡大縮小について不変にするため  $|u_1|$  で除算されているが、本研究の目的では相対的な大きさについても考慮するため、除算しない。結果として、閉領域  $i$  の特徴ベクトル  $\mathbf{f}_i$  を次のように計算する。

$$\mathbf{f}_i = (|u_0|, |u_1|, \dots, |u_{K-1}|), \quad (11)$$

すなわち、 $K$  個のフーリエ係数の絶対値からなるベクトルとする。閉領域  $i, j$  の対応付けのコスト  $a_{ij}$  は次の式で計算する。

$$a_{ij} = w_{trans}^{fourier} |\mathbf{t}_i - \mathbf{t}_j| + |\mathbf{f}_i - \mathbf{f}_j|, \quad (12)$$

ここで  $\mathbf{t}_i, \mathbf{t}_j$  はそれぞれ閉領域  $i, j$  の重心で (ただし画像サイズで正規化されている)、 $w_{trans}^{fourier}$  は重みである。パラメータについて、実験では文献 [11] と同様  $N = 64$  とし、 $K = 4$ 、 $w_{trans}^{fourier} = 0.5$  とした。

## 6. Shape Context

Shape Context [1] とは、ある点を原点とした極座標を考えたときの、動径の長さの対数と偏角についての2次元ヒストグラムである。対数距離を計算するとき、拡大縮小に対して不変にするため、予め各距離を平均距離で除算する。対数距離を使うため、原点に近い点の分布を細かく捉え、より遠い点の分布は粗く保持する。対数距離について  $N_{dist}$ 、偏角について  $N_{rad}$  の分割を行うと、計  $N_{bin} = N_{dist} N_{rad}$  個のビンからなるヒストグラムが得られる。二つのヒストグラム  $i, j$  の類似度  $C_{ij}$  を求めるには、カイ二乗を用いる。

$$C_{ij} = \frac{1}{2} \sum_{b=1}^{N_{bin}} \frac{|h_i(b) - h_j(b)|^2}{h_i(b) + h_j(b)}, \quad (13)$$

ここで  $h_i(b)$  と  $h_j(b)$  は、正規化されたヒストグラム  $i, j$  の  $b$  番目のビンに含まれる値である。

### 本研究での計算方法

本研究では、フーリエ記述子の場合と同様、閉領域の重心間の距離も考慮する。また、拡大縮小の影響を考慮するため、各閉領域での重心からの平均距離で除算することはない。閉領域  $i, j$  の対応付けのコスト  $a_{ij}$  は次の式で計算する。

$$a_{ij} = w_{trans}^{shape} |\mathbf{t}_i - \mathbf{t}_j| + C_{ij}, \quad (14)$$

ここで  $w_{trans}^{shape}$  は重みである。

パラメータについて、実験では文献 [1] と同様に、対数距離に関する分割数  $N_{dist} = 5$ , 偏角に関する分割数  $N_{rad} = 12$ ,  $w_{trans}^{shape} = 4$  とした。

## 7. 実験結果

比較用プログラムをC++, OpenGL, GLUT および GLUTI を用いて実装し、Intel Core i7-2820QM CPU 2.30GHz およびメモリ 3.2GB を搭載したノート PC 上で実験した。彩色済み画像から得られた、各閉領域に塗られる色を正解データとした。実験では、各形状記述子を用いて対応付けを行ったのち、あるフレームについてのみ色を塗り、色を自動で伝播させた際に、どれだけ正しい色を塗ることができたかを定量的に評価した。塗り始めるフレームは、最も閉領域の数が多いフレームとした。あるフレームで閉領域が新しく現れると、その閉領域は対応付けされずに残るため、できるだけ多く閉領域を含むフレームから塗り始める方が塗り残しを減らせるためである。

図 3, 4 および 5 に、実験結果を示す。画像は紙面の都合でトリミングしてある。楕円、フーリエ記述子および Shape Context を用いて対応付けを行い、誤って色が割り当てられた閉領域を赤、対応付けされずに色が割り当てられなかった閉領域を緑で表示している。「誤り」の数は「対応無」の数を含む。これらの結果について、各フレームの閉領域数に対して正しい色の割り当てられた閉領域数の割合を正解率とし、グラフにしたものが図 2 である。図 3, 4 および 5 のいずれにおいても、楕円よりもフーリエ記述子や Shape Context を用いた場合にわずかながらより高い正解率が得られた。

計算時間については表 1 の通りである。領域分割は 3 通りいずれも共通の処理で実現しており、計算時間はそれぞれの図で 0.34, 2.3, 8.1 秒であった。画像サイズに応じて領域分割にかかる時間は増加するものの、対応付けはいずれも非常に高速である。計算速度についても、楕円よりもフーリエ記述子や Shape Context を用いた場合の方が高速であるという結果が得られた。

図 2 のグラフについて考察する。いずれの結果においても、色付けを行ったフレームの前後で、急激に正答率が落ちている。色付けを行ったフレームは、最も閉領域数の多いフレームである。このようなフレームでは、髪や手や腕などで閉領域が遮られることで閉領域の数が増えている。遮蔽によって現れた複数の閉領域は、本来であれば分割される前の 1 つの閉領域と対応付けられるべきである。すなわち、閉領域に関して 1 対多の対応付けが必要となる。しかし、現在の 2 部グラフマッチングによる枠組みでは 1 対多の対応付けは行えず、対応付けられない閉領域が残り、結果として色付け精度が落ちてしまう。つまりこの結果

表 1 画像サイズとフレーム数ごとの対応付けの計算時間(秒). 計算時間はすべてのフレームに対する対応付けにかかった時間を表す。

図 (画像サイズ, フレーム数)	楕円	フーリエ記述子	Shape Context
図 3 (1024×768, 10 枚)	0.094	0.89	0.057
図 4 (3614×2053, 14 枚)	0.11	0.056	0.032
図 5 (3614×2053, 57 枚)	0.51	0.32	0.19

は、閉領域を 1 対 1 対応させる方式そのものについての限界を示しているものと考えられる。

## 8. まとめと今後の課題

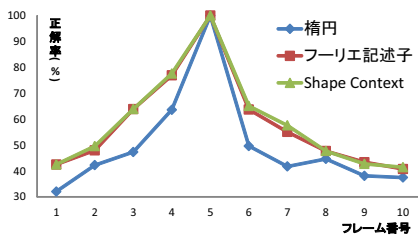
本研究では、現在の 2 次元手描きアニメの制作工程における彩色作業を半自動化するための手法について検討した。閉領域の形状記述子として楕円、フーリエ記述子および Shape Context の 3 つを用いて、それぞれ対応付けのコストを計算し、2 部グラフマッチングによって対応付けを行った。実験の結果、精度と計算速度の両方で、楕円よりもフーリエ記述子および Shape Context を用いた場合に良好な結果が得られた。しかしながら、現在の方法では閉領域に対して 1 対 1 の対応付けしか行えないため、遮蔽によって新たに閉領域が現れた場合に必要な 1 対多の対応が扱えず、遮蔽が多く発生する場合に精度が急激に低下する。今後は、遮蔽の影響をうまく扱える手法を開発したい。

## 謝辞

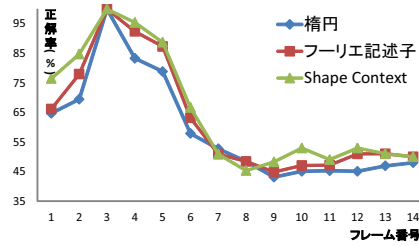
手描きアニメの画像データをご提供いただいた各社に深く感謝申し上げます。

## 参考文献

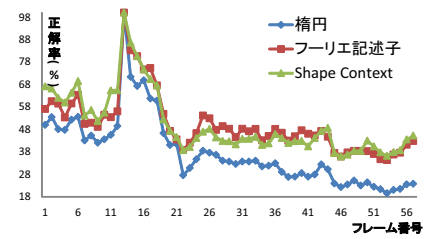
- [1] Belongie, S., Malik, J. and Puzicha, J.: Shape Matching and Object Recognition Using Shape Contexts, *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 24, No. 4, pp. 509–522 (2002).
- [2] Garcia Trigo, P., Johan, H., Imagire, T. and Nishita, T.: Interactive Region Matching for 2D Animation Coloring Based on Feature's Variation, *IEICE Transactions (E92-D)*, No. 6, pp. 1289–1295 (2009).
- [3] Kanamori, Y.: Region matching with proxy ellipses for coloring hand-drawn animations, *SIGGRAPH Asia 2012 Technical Briefs, SA '12*, pp. 4:1–4:4 (online), DOI: 10.1145/2407746.2407750 (2012).
- [4] Madeira, J. S., Stork, A. and Gross, M. H.: An approach to computer-supported cartooning, *The Visual Computer*, Vol. 12, No. 1, pp. 1–17 (1996).
- [5] Papadimitriou, C. and Steiglitz, K.: *Combinatorial optimization : algorithms and complexity*, Prentice-Hall (1982).
- [6] Qiu, J., Seah, H. S. and Tian, F.: Auto coloring with character registration, *Proceedings of the 2006 international conference on Game research and development, CyberGames '06*, pp. 25–32 (2006).
- [7] Sýkora, D., Ben-Chen, M., Čadík, M., Whited, B. and Simmons, M.: TexToons: practical texture mapping for hand-drawn cartoon animations, *Proceedings*



(a) 図 3 の例の正解率



(b) 図 4 の例の正解率



(c) 図 5 の例の正解率

図 2 図 3, 4 および 5 の例における自動色付けの正解率 (%) のグラフ. それぞれ最も閉領域数の多い 5 フレーム目, 3 フレーム目および 13 フレーム目に色を指定しているため, それらのフレームでの正解率は 100%となっている.

	1 フレーム目	4 フレーム目	6 フレーム目	10 フレーム目
入力画像				
	閉領域数: 134	閉領域数: 143	閉領域数: 149	閉領域数: 152
楕円				
	誤り: 87 (対応無: 6)	誤り: 42 (対応無: 0)	誤り: 67 (対応無: 0)	誤り: 90 (対応無: 12)
フーリエ記述子				
	誤り: 76 (対応無: 5)	誤り: 33 (対応無: 0)	誤り: 54 (対応無: 0)	誤り: 91 (対応無: 14)
Shape Context				
	誤り: 72 (対応無: 11)	誤り: 28 (対応無: 0)	誤り: 54 (対応無: 0)	誤り: 82 (対応無: 22)

図 3 10 フレームからなるシーケンスを用いた実験結果. 赤は誤った色の割り当てられた閉領域, 緑は対応付けされず色が割り当てられなかった閉領域を表す. 5 番目のフレームに色付けし, 第 1, 4, 6, 10 フレーム目の結果のみ抜粋. ©株式会社セルシス.

of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering, NPAR '11, pp. 75–84 (online), DOI: 10.1145/2024676.2024689 (2011).

[8] Šýkora, D., Dingliana, J. and Collins, S.: As-rigid-as-possible image registration for hand-drawn cartoon animations, *Proceedings of the 7th Interna-*

*tional Symposium on Non-Photorealistic Animation and Rendering*, NPAR '09, pp. 25–33 (online), DOI: 10.1145/1572614.1572619 (2009).

[9] Šýkora, D., Dingliana, J. and Collins, S.: LazyBrush: Flexible Painting Tool for Hand-drawn Cartoons, *Computer Graphics Forum*, Vol. 28, No. 2, pp. 599–608 (2009).

	1 フレーム目	2 フレーム目	4 フレーム目	14 フレーム目
入力画像				
	閉領域数: 68	閉領域数: 59	閉領域数: 66	閉領域数: 50
楕円				
	誤り: 23 (対応無: 13)	誤り: 15 (対応無: 4)	誤り: 13 (対応無: 4)	誤り: 28 (対応無: 3)
フーリエ記述子				
	誤り: 24 (対応無: 4)	誤り: 13 (対応無: 0)	誤り: 5 (対応無: 0)	誤り: 26 (対応無: 0)
Shape Context				
	誤り: 17 (対応無: 8)	誤り: 9 (対応無: 0)	誤り: 3 (対応無: 0)	誤り: 26 (対応無: 2)

図 4 14 フレームからなるシーケンスを用いた実験結果. 赤は誤った色の割り当てられた閉領域, 緑は対応付けされず色が割り当てられなかった閉領域を表す. 3 番目のフレームに色付けし, 1, 2, 4, 14 フレーム目の結果のみ抜粋. ©2006 二馬力・GNDHDDT.

- [10] Yang, M., Kpalma, K. and Ronsin, J.: A Survey of Shape Feature Extraction Techniques, *Pattern Recognition*, pp. 43-90 (2008).
- [11] Zhang, D. and Lu, G.: A comparative Study of Fourier Descriptors for Shape Representation and Retrieval, *Proc. of 5th Asian Conference on Computer Vision (ACCV)*, Springer, pp. 646-651 (2002).


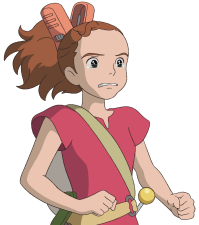






















	1 フレーム目	11 フレーム目	12 フレーム目	14 フレーム目	15 フレーム目	57 フレーム目
入力画像						
	閉領域数: 70	閉領域数: 101	閉領域数: 107	閉領域数: 119	閉領域数: 98	閉領域数: 84
楕円						
	誤り: 40 (対応無: 0)	誤り: 11 (対応無: 0)	誤り: 56 (対応無: 0)	誤り: 24 (対応無: 0)	誤り: 29 (対応無: 0)	誤り: 57 (対応無: 11)
フーリエ記述子						
	誤り: 30 (対応無: 0)	誤り: 47 (対応無: 0)	誤り: 47 (対応無: 0)	誤り: 20 (対応無: 0)	誤り: 19 (対応無: 0)	誤り: 48 (対応無: 5)
Shape Context						
	誤り: 22 (対応無: 2)	誤り: 32 (対応無: 0)	誤り: 35 (対応無: 0)	誤り: 16 (対応無: 0)	誤り: 19 (対応無: 0)	誤り: 49 (対応無: 27)

図 5 57 フレームからなるシーケンスを用いた実験結果. 赤は誤った色の割り当てられた閉領域, 緑は対応付けられず色が割り当てられなかった閉領域を表す. 13 番目のフレームに色付けし, 1, 11, 12, 14, 15, 57 フレーム目の結果のみ抜粋. ©2010 GNDHDDTW.