

キャラクターの外見的特徴量の計測実験

高松耕太[†] 嶋津恵子^{††}

我々は、人物を対象とした日本の文化財がデフォルメ手法を利用していることに注目した。デフォルメの程度や傾向をはかることで、それぞれの特徴を工学的に特定することが可能になると考えた。今回我々は、代表的な画像処理技術のひとつであるラベリング処理を応用したデフォルメ度計測手法を提案する。さらにこの方法を利用して、実際にキャラクター画像のデフォルメ度を計測した実験結果を報告する。

Experimental study of measuring digital character features

Kota Takamatsu[†] and Keiko Shimazu^{††}

We are interested that Japanese many cultural properties were employed a methodology of deformation. There is a possibility that measuring their deformation amount leads that their feature quantity is identified. In this experimental study, we applied labeling method, one of the most popular image processing technologies, to produce 2 types of features. We experimented to execute 57 digital images' features using our method.

1. はじめに

日本の創作者は、縄文時代の土偶から、江戸時代の浮世絵、さらには現代のアニメキャラクターに至りデフォルメ手法を活用しているように見受けられる[1].

今回我々は、デジタル表現されたキャラクター画像を対象に、「省略」と「誇張」のデフォルメ度を計測する手法を考案し、これを使って外見的特徴量の計測を試みた。

[†]慶應義塾大学大学院 システムデザイン・マネジメント研究科附属
システムデザイン・マネジメント研究所
SDM Research Institute / Keio Univ.
^{††}慶應義塾大学先端研究センター
Keio Advanced Research Centers

2. 先行・関連研究

手塚治虫は、「漫画」のキャラクター画像のデフォルメの基本要素は、「省略」と「誇張」そして「変形」だとした[2]. 「省略」は、キャラクターを描く線の数を減らす操作である。これによりキャラクターの構成要素の単純描画化が実現される。「省略」度が高いと、キャラクターを描く線の数が少なくなり、描画は単純になる。「誇張」は、キャラクターを描く線を太くしたり、特定の構成要素を大きくしたりすることで、意図的にバランスを崩す操作である。頭部は目などの個性を表す構成要素が多いため、「誇張」を適用することが多い。「省略」と「誇張」の操作で、モチーフとなった対象と大きく異なる輪郭や外形になる。これが「変形」、つまりデフォルメである。

河谷らはアニメのキャラクター画像の頭部を対象に、「誇張」度を計算するアルゴリズムを開発した[3][4]. また、宇津野らは白黒の刺繍デザイン画像を対象に「省略」「誇張」度の抽出手法を開発し、感性情報との関連付けを行っている[5].

河谷らに対し、我々は全身を対象に「省略」度と「誇張」度の両方を計測し、デフォルメ度を抽出する手法を考案した。また、白黒画像への変換前処理方法も考案し、カラー画像に対しても、計測可能にした。

3. 外見的特徴の抽出方法

我々が考案した「省略」度と「誇張」度で、画像の特徴量をデフォルメ度で測る方法は3つの操作から成り立つ。具体的には、(i)画像処理における一般的な前処理を行った後、(ii)ラベリング処理による「省略」度の計測と、(iii)頭身の低さによる「誇張」度の計測であり、この(ii)と(iii)の2値で、デフォルメ度を表現する(Fig1).

3.1 前処理：画像の白黒線画処理

電子的に描かれた画像は、正方形(まれに長方形)の形状をした最小構成単位(画素)の集まりで表現される[6]. 今回、前処理として、対象となる画像ごとに(ア)二値化した結果と、(イ)エッジ検出した結果を、(ウ)合成した。二値化は、元の画像を白黒の単純な線画に変換する代表的な手法である。構成するそれぞれの画素の明度を測り、それらが閾値(域値)を超えたとき黒

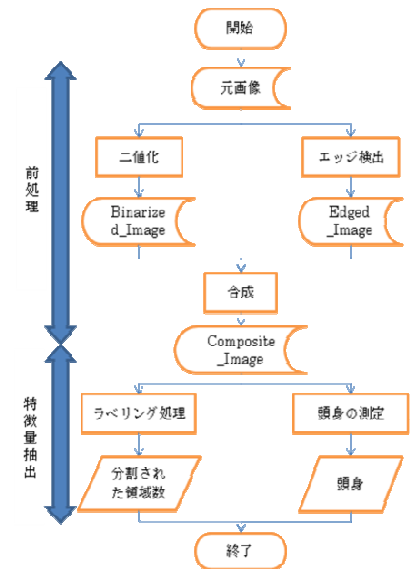


Fig1. 外見的特徴の抽出手順

色，そうでないとき白色を示す RGB に変換する(Binarized_Image).

一方，エッジ検出も，元の画像を白黒の単純な線画に変換する代表的な手法である．構成するそれぞれの画素の明度と隣接画素のそれを測る．それらの差が閾値（域値）を超えたとき黒色，そうでないとき白色を示す RGB に変換することで実現している(Edged_Image).

二値化もエッジ検出も，画像処理に十分な情報が概ね抽出されるが，欠損値が発生することも多い[7][8]．これに対し，我々はそれぞれの操作の結果画像を合成することで，補正を行った．二値化処理した結果(Edged_Image)とエッジ検出した結果(Binarized_Image)の合成は，両画像の同じ地点の画素を比較し，一方もしくは両方が黒色のとき黒色，そうでないとき白色を示す RGB に変換した(Composite_Image)．次節以降に示す計測には Composite_Image を対象に行った．

3.2 「省略」度の測定：描画の粗さ

デフォルメされた画像の対象モチーフからの「省略」を表す尺度として，線の量や，線によって分断された領域数などが挙げられる．我々は，後者の代表であるラベリング処理技術を採用した[6]．これは，任意の画素(pixel_0)に対し，それらを直接囲む 8 個の画素のうち，上下左右の 4 つ(pixel_1...pixel_4)を四近傍とし，これらに対して行う次のような操作である． pixel_0 が， pixel_1...pixel_4 のいずれか，もしくはすべてと同一の色である（連結）とき，それらの画素すべてに同じラベル（整数）を割り当てる．連結が終了した時点で，特定の整数の割り当てが終了し，別の連結集合には 1 を加えた整数を用いる．従って，ラベリング処理の結果，割り当てられた整数のうち，最大値である数値がその画像の線で分断された領域数となる．Fig1 の例では，分断された領域の数は 16 となる．

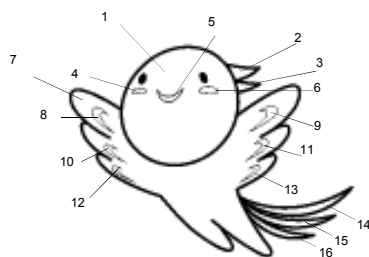


Fig1 分断された領域の計測例（はばタン[9]）

Table1 は 2 つの画像（地デジカ[10]とリラックマ[11]）を対象に，モチーフの写真とデフォルメした画像を，ラベリング処理の分断された領域数で比較した

	モチーフ	イラスト
サンプル 1	カモシカ:488	地デジカ:13
サンプル 2	クマ:569	リラックマ:12

Table1 ラベリング処理アルゴリズムを使って分断された領域の数を計算した例

ものである．この例の様に，ラベリング処理は，描画の粗さを定量的に示せることが特徴である．

3.3 「誇張」度の測定：頭身の低さ

「誇張」を表す尺度として，縦横比や曲線の直線化率，および特定部分の削除や拡大が挙げられる．特に頭部は，表情やパーソナリティを表現するのに適した素材が多くあるため，多くのキャラクター画像では，対象モチーフと比較し，全身に対する頭部の大きさの比率が大きい．そこで，今回我々は，「誇張」の尺度として，頭身を用いた．我々は全身と頭のそれぞれの大きさを次のように定義し，人手により測定を行った．

- ・頭の大きさ：頭頂から顎までの距離（顎が無い場合は，下唇までの距離）
- ・全身：頭頂から足の裏までの距離（足が無い場合は，体の最下部までの距離）
- ・頭身：全身を頭の大きさで割った値

なお，頭に付属している突起物（例えば，キャラクターが被っている兜の角など）は，計測に含めない．

3.4 デフォルメ度の特定

デフォルメ度である「変形」度の大きさは「省略」度と「誇張」度で実現される．そこで 3.2 節，および 3.3 節の方法で計測される「省略」度 (S:simplified) と「誇張」度 (E:emphasis) の組 (S,E) でデフォルメ度を特定できると考えた．

また，分断された領域数の数が小さくなると実際にデフォルメ画像の「省略」された印象が高まるため，「省略」度 (S) は計測値の逆数を用いた．そして，「誇張」度 (E) にも同様の理由で計測値の逆数を用いた．

4. 実験

57 のキャラクター画像を対象にデフォルメ度の計測実験を行い，人が目で見たときの印象と照合することで，我々の方法の有用性を検証した．

Table2 は、画像のデフォルメ度を、「省略」度(S)と「誇張」度(E)で算出した結果(部分)である。また、Fig4 は、X 軸に「誇張」度(E), Y 軸に「省略」度(S)を設定し、これを二次元グラフ上にプロットしたものである。特にデフォルメ度の高い結果になった①～④を Fig5 に示し、デフォルメ度が低い結果になった⑤～⑧を Fig6 に示す。前者は単純な線画であり、後者は複雑な線画である。

	名前	「省略」度(S)	「誇張」度(E)
①	ひこにゃん[12]	0.056	0.67
②	やちにゃん[13]	0.037	0.68
③	パパたこ[14]	0.034	0.73
④	さがわん もも[15]	0.053	0.59
⑤	アンジェリーク[16]	0.004	0.13
⑥	天蓬元帥[17]	0.007	0.13
⑦	原村 和[18]	0.008	0.16
⑧	水野 楓[19]	0.006	0.15

Table2 デフォルメ度の計測結果 (部分)



Fig5. デフォルメ度が高い画像例
(左から①ひこにゃん[12], ②やちにゃん[13], ③パパたこ[14], ④さがわん もも[15])

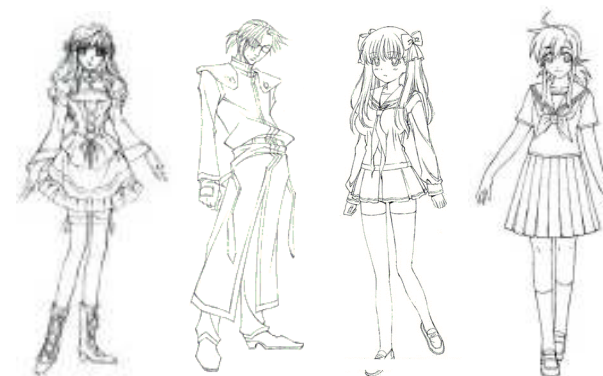


Fig6. デフォルメ度が低い画像例
(左から⑤アンジェリーク[16], ⑥天蓬元帥[17], ⑦原村 和[18], ⑧水野 楓[19])

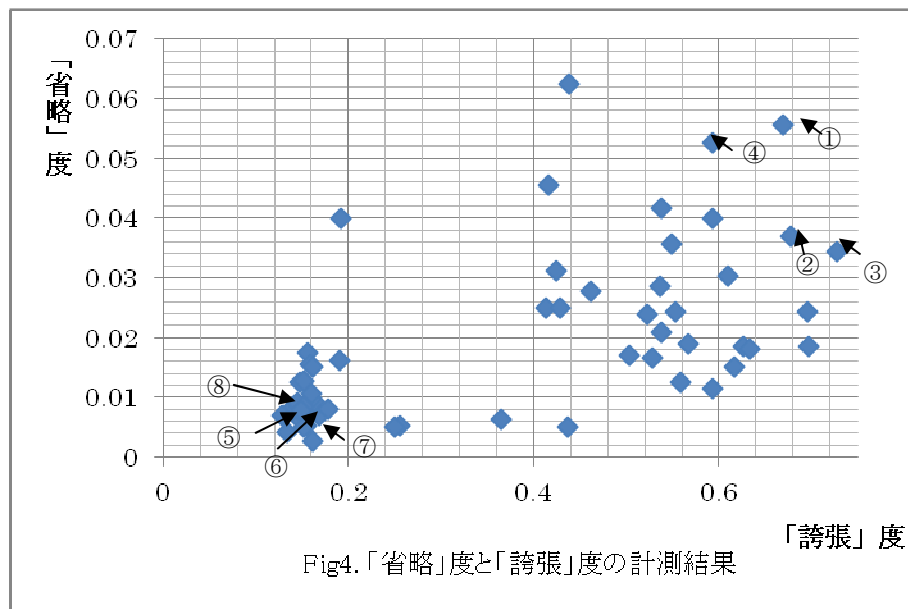


Fig4. 「省略」度と「誇張」度の計測結果

5. 考察

今回我々は、デジタル画像のデフォルメ度を「省略」度と「誇張」度の2つの計測結果を用いて特定する方法を考案した。

この2つの数値の組でデフォルメ度が高いと判定された画像は、見た目の印象も同様にデフォルメ度が高いと言える。その一方で、デフォルメ度が低い数値とされているものも、目などの特定部分が、大きく誇張された印象を与えていることがわかった。

今後、実在する人間等の写真画像の数値を計測し、それを基準値として、デフォルメ傾向を測る工夫を試みる予定である。

6. まとめ

今回我々は、キャラクターを対象に「省略」度と「誇張」度を採用し、特徴量の測定方法を開発した。また、これを利用し、実際に57体の特徴を測定した。今後はより精度の高いデフォルメ度を抽出するため、実在するモデルからのデフォルメ傾向の計測方法を検討する。

謝辞 論文の作成にご協力頂いた皆様に、謹んで感謝の意を表する。

参考文献

- 1) <http://muse.main.jp/arts/00153.html> (2010.12.19 参照)
- 2) 手塚治虫, マンガの描き方—似顔絵から長編まで, 光文社, 1996年
- 3) 河谷大和, 柏崎礼生, 高井昌彰, 高井那美, アニメキャラクターの特徴抽出に基づくアニメ度の評価, IPSJ SIG Notes 2008(80), 35-38, 2008年
- 4) 河谷大和, 柏崎礼生, 高井昌彰, 高井那美, アニメにおける人物顔画像の萌え因子特徴評価と検索分類システムへの応用, ITE Technical Report 34(6), 113-118, 2010年
- 5) 宇津野直木, 近藤邦雄, 佐藤尚, 島田静雄, 画像特徴量を用いたデザイン画像の感性情報抽出, 全国大会講演論文集 第52回平成8年前期(2), 73-74, 1996年
- 6) 村上伸一, 画像処理工学, 東京電機大学出版局, 2004年
- 7) 三浦弘之, 翠川三郎, 建物GISデータの更新を目的とした高分解能衛星画像からの建物の自動検出手法, Journal of social safety science (5), 37-44, 2003-11
- 8) 村形明, 恩田憲一, 小沢慎治, 多値文書画像における適応的二値化手法, Technical report of IEICE. PRMU 99(649), 9-16, 2000-02-22
- 9) <http://web4.pref.hyogo.lg.jp/habatan2006/> (2010.12.19 参照)
- 10) <http://www.nab.or.jp/chidejika/> (2010.12.19 参照)
- 11) <http://www.san-x.co.jp/relaxuma/top.html> (2010.12.19 参照)
- 12) <http://hikone-hikonyan.jp/> (2011.2.27 参照)
- 13) <http://yachinyan.shiga-saku.net/> (2011.2.27 参照)
- 14) <http://www.taco-ferry.com/> (2011.2.27 参照)
- 15) <http://www.sagawa-artmuseum.or.jp/> (2011.2.27 参照)
- 16) <http://www.gamecity.ne.jp/neoromance/tv/neo/> (2011.2.27 参照)
- 17) <http://saiyuki-gaiden.com/> (2011.2.27 参照)
- 18) <http://www.saki-anime.com/> (2011.2.27 参照)
- 19) <http://www.tbs.co.jp/anime/nyankoi/> (2011.2.27 参照)