

2010年度 卒業論文

キャラクター画像を利用した3次元
キャラクターモデルの顔形状の変形に関する研究

指導教員：渡辺 大地 講師

メディア学部 ゲームサイエンスプロジェクト
学籍番号 M0107351
成田 悟

2010年度 卒業論文概要

論文題目

キャラクター画像を利用した3次元
キャラクターモデルの顔形状の変形に関する研究

メディア学部

学籍番号：M0107351

氏名

成田 悟

**指導
教員**

渡辺 大地 講師

キーワード

キャラクター画像, 特徴点, アバター, キャラクターメイキング, 自動生成

近年、ゲームはハードウェアの向上に伴い、キャラクターに関するユーザーへの自由度が高くなっている。そのなかでも、顔の形や目の大きさなどの容姿をいくつかあるパラメータを調整することで自分の理想とするキャラクターの容姿をつくることのできるシステムであるキャラクターメイキングがある。キャラクターメイキングは、自由度の高さやユーザーの個性を反映させやすいというメリットがあるが、パラメータを調節するために時間がかかってしまうというデメリットがある。パラメータ調節の時間短縮のために、目の大きさなどの各パーツのパラメータを設定しておいたデータであるテンプレートデータをいくつか用意しておくというものがある。ユーザーは、複数あるテンプレートデータの中から自分の理想とするキャラクターに近い形状のものを選択するだけで、ベースとなる形状になるようにパラメータが設定されるので、パラメータの微調整を行うだけで自分の理想とするキャラクターを作ることができる。だが、この方法では、そのテンプレートデータを作るために誰かがパラメータを調節しなければならない。また、自分の理想とするキャラクターの容姿をしたテンプレートデータがない場合は、結局は自分でパラメータを調節しなければならない。

キャラクター画像は比較的入手しやすく、自分の理想とするキャラクターを探しやすい。そして、キャラクター画像は、ゲームで使われるキャラクターと近い形状をしている。そこで本研究では、キャラクター画像を利用してパラメータを利用したキャラクターメイキングにおけるパラメータを自動的に調節することを目的とした。具体的には、ユーザーはキャラクター画像に対して特徴点を指定する。その特徴点を元にしてパーツごとの大きさの比率や位置関係を算出していく。そして、それらの関係をパラメータに反映させることで、自分の理想とするキャラクターを作る。

最後に、アンケートを用い本手法の検証を行った。元にしたキャラクター画像との比較では変形できる部位の不足や髪と色を考慮しなかったため良い結果を得ることができなかった。こういったものはキャラクターの印象に影響してくるので、必要性が分かった。やりやすさに関しても、ユーザーインターフェイスの問題から良い結果を得ることができなかった。検証の際、本手法のユーザーインターフェイスが使いにくく設計されていたので、本手法を使いにくく感じる人がいた。

目次

第1章	はじめに	1
1.1	背景と意義	1
1.2	論文構成	6
第2章	提案手法	7
2.1	変形を行う部位	7
2.2	利用する画像	9
2.3	利用する画像の特徴点	10
2.4	特徴点の分析と反映	12
第3章	システムの検証と考察	21
3.1	検証方法	21
3.2	実装	22
3.3	結果と考察	22
3.4	今後の展望	24
第4章	まとめ	27
	謝辞	29
	参考文献	30

第 1 章

はじめに

1.1 背景と意義

近年、ゲームはハードウェアの向上に伴い、キャラクターに関するユーザーへの自由度が高くなっている。その一例として、アバター [1] がある。アバターは、仮想空間における自分の分身として、コミュニケーションツールで利用出来るキャラクターやサービスのことである。ヘアスタイルや装飾品などのパーツを選ぶことで自分だけのキャラクターを作ることができるので、ユーザーの個性を表すことができる。例として、パーツを組み合わせることで、自分の理想とする容姿や自分の好きなキャラクターの容姿のアバターをつくることができる。そのため、そのアバターを見れば、そのユーザーがどのような容姿を好んでいるか、どんな作品が好きかなどがわかる。

その中でも、特に自由度が高くなったといえるキャラクターに関するシステムとして、キャラクターメイキングがある。キャラクターメイキングとは、予め用意されたゲームシステムを利用して、顔、目、ヘアスタイルなどの自分で操作するキャラクターの容姿を自分で設定・変更するシステムのことを指す。ゲームによってはこのようなシステムのことを、キャラクターエディットなどのように違う用語を用いている場合がある。そのため、先ほど述べたようなことができるシステムであれば、本研究ではキャラクターメイキングと定義し、同様に扱うものとする。本研究では、このキャラクターメイキングというシステムに着目した。キャ

ラクターメイキングが一般的になる前のゲームでは、予め制作者によって決められたキャラクターを操作してゲームを遊んでいた。キャラクターメイキングが行えるようになるようになると、キャラクターメイキングを利用することで、自分の好きな容貌のキャラクターを作りそのキャラクターを操作して遊ぶことができるようになった。キャラクターメイキングは、主に複数の人数で遊ぶことのできるアクションゲームやMMORPGといったジャンルのゲームで多く見ることができる。決められたキャラクターを操作するゲームと違い、自分で容姿を変更することができるので、プレイヤーの個性が出やすくキャラクターに対するユーザーの思い入れも高くなる。しかし、ゲームを始める前にキャラクターを作る作業を行うため遊ぶ前に時間がかかる。キャラクターメイキングには大きくわけて2通りの方法がある。1つは、制作者によって用意されたパーツを組み合わせることで、キャラクターをつくり出す方法である。例として「ファンタシースター」シリーズ [2] がある。もう一方は、パラメータを調整してキャラクターをつくり出すものである。本研究では、後者のパラメータを利用したキャラクターメイキングを研究対象とする。パラメータを利用してキャラクターメイキングを行う方法は、目の大きさ、眉の位置、顎の形状などのキャラクターの容姿を、パラメータの値を調節することで、それぞれのパーツの大きさや位置などを変えてキャラクターを作成するものである。「ARGO ONLINE」 [3] では、顔の形、目の位置、鼻の大きさなどを変更することができる。図 1.1 はその例を示している。パラメータを利用したキャラクターメイキングはパーツを利用したキャラクターメイキングと比較して、自由度が高いという反面、パラメータ調節に時間がかかるという問題がある。パーツ選択を利用したキャラクターメイキングであれば自分の理想とする容姿のパーツを選ぶだけであり、つくる時間はあまりかからない。しかし、パラメータを利用したキャラクターメイキングではパーツの位置やサイズを細かに設定するため、自分の理想とする容姿に近づけるだけでも時間がかかる。例として、「パーフェクトワールド-完美世界-」 [4] では、顔の輪郭の形、顔の横幅・縦幅、目尻の角度、眼球の大きさなど、100個以上のパラメータを利用して、パラメータを利用

したキャラクターメイキングを行うことができる。しかし、パラメータが多ければ多いほど、それだけのパラメータを自分で調節しなければならず、自分の理想に近い容姿のキャラクターを大まかにつくるだけでも時間がかかる。その上、パラメータを細かく調整するともなれば、さらに時間を必要とする。

これらの問題に対しての解決方法として、「The Tower of AION」[5]では、作業時間の短縮のためにテンプレートデータを利用している。制作者が、予め顔や体格などのパラメータを設定しておいたデータであるテンプレートデータをいくつか用意しておくというものである。そのため、ユーザーは複数あるテンプレートデータの中から自分の理想とするキャラクターに近い形状のものを選択するだけで、ベースとなる形状になるようにパラメータを設定できる。その後、ユーザーはパラメータの微調整を行い、自分の理想とするキャラクターを創り上げることでパラメータ調節にかかる作業時間の短縮を行っている。また、「パーフェクトワールド-完美世界-」では、先ほど述べたテンプレートデータを自分で作成・保存することができる。そのため、気に入った容姿をしたキャラクターが作ることができた場合、そのパラメータを保存することができる。そのため、次回以降のキャラクターメイキングの際、同一容姿のキャラクターやそれに近いキャラクターを作る場合、ベースとなる形状の作成がテンプレートデータを選ぶだけになるので、作業時間を減らすことができる。また、インターネットを利用してそれらテンプレートデータをやり取りすることで、他の人が作ったキャラクターを再現できる。そのため先程のあらかじめ用意されたテンプレートデータを選択する方法と比較して、自分の理想とする形状にしやすくなる。しかし、テンプレートデータの中に自分の理想とする容姿のデータがなければ、結局は自分でパラメータを調節することになる。また、テンプレートデータを作るために誰かがパラメータ調節を行わなければならない。

梁らの研究[6]では、顔の画像から抽出した特徴点を元して類似したパーツを選び組み合わせることで、似顔絵を作成するというものがある。しかし、予め用意されたパーツを選択するので、パーツを利用したキャラクターメイキングでは利

用することはできるが、パラメータを利用したキャラクターメイキングでは利用することができない。

武本らの研究 [7] では、顔形状を 3 次元データとして計測したデータの中から特徴点を選択し、平均的な顔形状モデルと比較することで、モデル形状やテクスチャーに誇張表現を施したモデルを作り出すことができる。しかし、モデルを作るためには特殊な機器を用いて実際の顔の点群データを取る必要がある。一般的な家庭にはそのような機器はないので、本研究で想定しているキャラクターメイキングにおいて利用することはできない。

五十嵐らの研究 [8] や Olga らの研究 [9] では、簡単な絵を描くことで簡易的な 3 次元モデルを作ることができる。しかし、新規にモデルを作るため、本研究の想定するキャラクターメイキングで使うことは困難である。

吉田らの研究 [10] や小野らの研究 [11] では、アニメーションを利用してモデルを自動生成する事ができる。しかし、映像の全てのフレームやそれぞれのアングルの画像に対して、特徴点を指定しなければならない。また、アニメーション映像を用意する必要がある。そのため映像の入手のしやすさや手間を考えると、本研究の想定するキャラクターメイキングで利用することは困難である。

森らの研究 [12][13][14] では、正面と側面の 2 枚の画像を利用して 3D デフォルメキャラクターを生成している。また今尾らの研究 [15] では、一般的な人体の形状のモデルを正面と側面の 2 枚のシルエット画像を利用して変形し、仮想空間での服の試着手法を提案している。これらの手法では、3次元測定用の装置や機材を用いず、また数多くのモデルを作る必要もなく目的のモデル形状を作成することができる。しかし、キャラクターメイキングをこの手法を利用して行う際、自分が理想とするキャラクターの真正の映っている画像と側面の映っている画像の 2 枚を集める必要がある。キャラクター画像で側面を正確に向いている画像は少ない。そのため本研究で想定しているキャラクターメイキングでは利用することが困難である。

森の研究 [16] では、2 枚の画像とデータベースを利用してモデルを作成する方法

がある。この手法では、予め3次元測定用の装置や機材を利用してモデルデータを幾つかつくっておき、データベースとして保存しておく。そこから、2枚の画像を利用して、そのデータベースを利用してモデルをつくる。しかし、この方法では予め大量のデータを必要とする。新規にモデルを作るので、本研究の想定しているキャラクターメイキングでは利用することができない。

それらを踏まえ本研究ではパラメータを自動で調節する手法を提案する。パラメータ調節を自動で行うことで、自分の理想のキャラクターを素早く作ることができるからである。自分の理想とするキャラクターには、アニメーションや漫画などのキャラクターが映っているキャラクター画像に注目した。キャラクター画像は、Webを探せば自分の好みのキャラクター画像を簡単に見つけることができるからである。本研究ではキャラクター画像を利用してキャラクターメイキングのパラメータを自動的に調節する手法を提案する。なお、今回のキャラクターは人物のものに限る。これにより、ユーザーはパラメータ調節にかかる時間を減らすことができる。処理の大まかな流れとして、ユーザーはキャラクターの画像に特徴点を設定する。その特徴点をもとに各特徴点間の距離を利用して、目や耳などのパーツの位置関係やパーツの大きさを求める。パーツの大きさの比率をそれぞれ求め、その比率データをもとにして位置関係や大きさの判定を行い、パラメータを自動で調節する。また、本手法を実装したプログラムを用い、本研究の有用性を検証した。

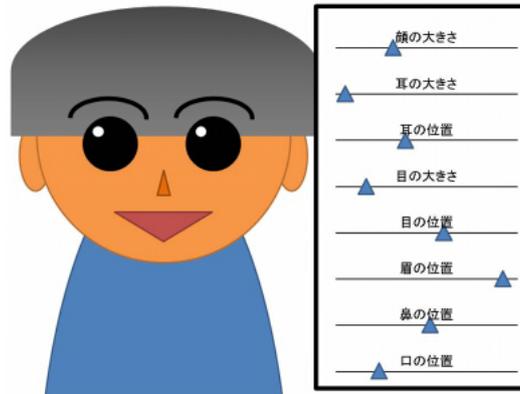


図 1.1: パラメータを利用したキャラクターメイキングのイメージ図)

1.2 論文構成

本論文の構成は次のとおりである。第2章では、キャラクター画像を利用したパラメータの自動設定の具体的な手法について述べる。第3章では、本手法の有効性についての検証を行う。第4章では、全体のまとめと今後の課題を述べる。

第 2 章

提案手法

本章ではキャラクターメイキングの効率化を図るシステムを提案する。まず、本研究ではモデルのどの部位を変形するかを述べる。次に、本研究で扱うキャラクター画像について述べ、そのキャラクター画像の特徴点について述べる。最後に、それら特徴点をどのように利用してパラメータに対応させるのかを述べる。

2.1 変形を行う部位

本研究では次に示すモデルの部位の変形を行う。顔に対するモデルの変形はパラメータを変更することで、顔の縦幅と横幅の比率を調整することができる。図 2.1 における青い線の縦線は顔の縦幅、横線は顔の横幅をそれぞれ表している。目に対するモデルの変形はパラメータを変更することで、目の顔の横幅に対する位置の調整と目の大きさを調整することができる。図 2.2 における赤い点は目の位置を表し、青い線で囲まれた部分は目の大きさを表している。耳に対するモデルの変形はパラメータを変更することで、耳の縦方向に対する位置の調整、耳の大きさを調整することができる。図 2.3 における赤い点は耳の位置を表し、青い線は耳の大きさをそれぞれ現している。眉に対するモデルの変形はパラメータを変更することで、眉の位置の調整をすることができる。図 2.4 における赤い点は眉の位置を表す。鼻に対するモデルの変形はパラメータを変更することで、鼻の位置の調

整をすることができる。図2.5における赤い点は鼻の位置を表している。口に対するモデルの変形はパラメータを変更することで、口の位置の調整をすることができる。図2.6における赤い点は口の位置を表している。

ゲームによってはパーツ選択を利用して、キャラクターに刺繍を入れたり、獣耳やアクセサリなどの装飾を施すことやヘアスタイルを自由に決められるが、これらはパラメータを利用したものではないので、本研究では扱わない。それに加えて、本研究ではヘアスタイルと瞳などの色も扱わない。また、あるパラメータを変化させると一部のパラメータの値が変化するものもある。例として、目の大きさを変えたとき、それに合わせて目の位置を微調整するというものである。このようなシステムを利用している作品として「Demon's Souls」 [17]がある。このようなものはパラメータ調整を支援するシステムではあるが必ず必要なものであるというものではないので、本研究では扱わない。



図 2.1: 顔の変形する部分

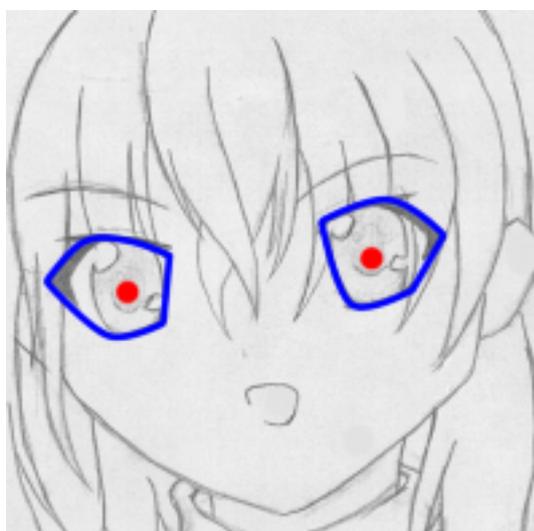


図 2.2: 目の変形する部分

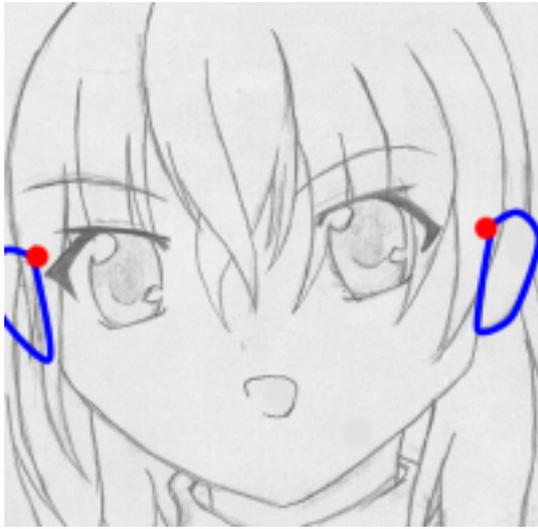


図 2.3: 耳の変形する部分



図 2.4: 眉の変形する部分

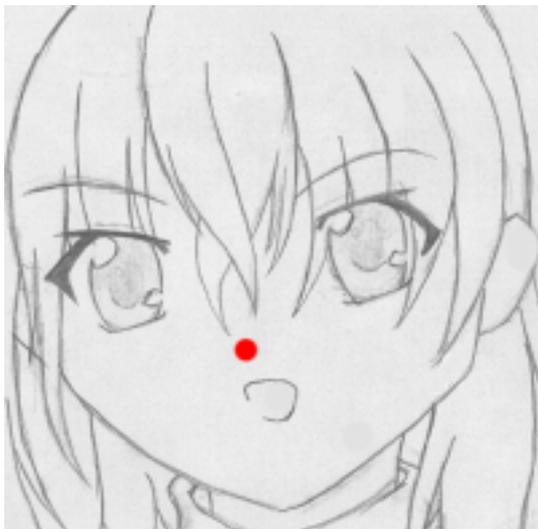


図 2.5: 鼻の変形する部分

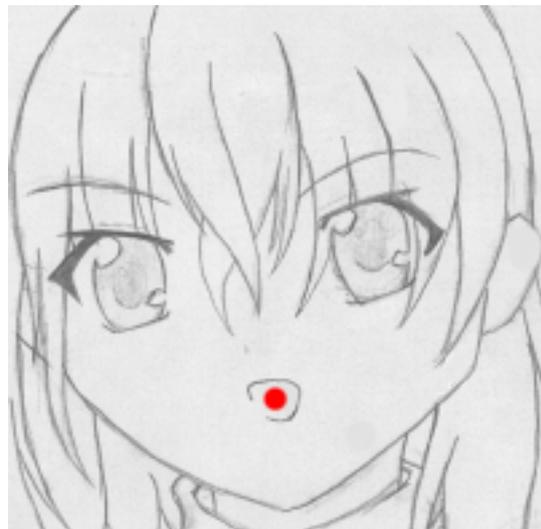


図 2.6: 口の変形する部分

2.2 利用する画像

本研究では自分の理想とするキャラクター像として、キャラクター画像を利用する。キャラクター画像は、Web を探せば簡単に見つかるので利用しやすいから

である。しかし、キャラクター画像と言っても、キャラクターは全て同じ方向を向いているわけではなく、画像によって向く方向が異なる。横を向いているキャラクター画像では、顔のパーツの位置関係や大きさといったものが判断しづらい。また、アングルが上から見下ろしているようなキャラクターの俯瞰画像も同様に、特徴の判別をしづらくなる。そのため本研究では、正面を向いたキャラクター画像を利用する。

2.3 利用する画像の特徴点

キャラクターの特徴の判定にはユーザーが画像に対して指定した特徴点を利用する。特徴点は指定する量を多くすれば、より詳細なデータを得ることができるがそれだけ手間がかかってしまう。よって、本研究で利用する特徴点は、輪郭に3点、左右の目にそれぞれに5点、左右の耳それぞれに2点、左右の眉それぞれに1点、鼻に1点、口に1点の合計21点を利用する。

次に、それぞれの部位に対して特徴点の指定箇所を述べる。輪郭の特徴点は、顎の先端に1箇所、左右のこめかみに1箇所ずつ指定する。図2.7は、左右のこめかみに対してそれぞれ1点ずつ、顎に1点指定したものである。目の特徴点は、左右の目に対して上瞼の両端に2箇所、上瞼の中央に1箇所、下瞼の両端に2箇所指定する。図2.8は、左右の目の上瞼の両端にそれぞれ2点、上瞼の中央にそれぞれ1点、下瞼の両端にそれぞれ2点指定したものである。耳の特徴点は、左右の耳に対して上下の両端の点を指定する。図2.9は、左右の耳の上下の両端にそれぞれ1点ずつ指定したものである。眉の特徴点は、左右の眉に対して眉間に近い端の点を指定する。図2.10は、左右の眉の眉間に近い端の1点を指定したものである。鼻の特徴点は、鼻先の点を指定する。図2.11は、鼻先の点を指定したものである。口の特徴点は、口の中央の点を指定する。図2.12は、口の中央の点を示したものである。



図 2.7: 輪郭の特徴点を指定する場所



図 2.8: 目の特徴点を指定する場所



図 2.9: 耳の特徴点を指定する場所

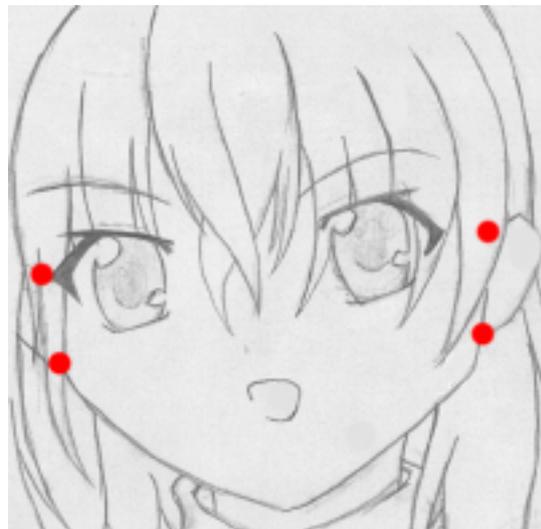


図 2.10: 眉の特徴点を指定する場所



図 2.11: 鼻の特徴点を指定する場所



図 2.12: 口の特徴点を指定する場所

2.4 特徴点の分析と反映

2.3節で指定した特徴点を元にして、それぞれのパーツの大きさや位置関係を算出する。特徴の分析には位置関係を利用する。位置関係には任意の特徴点2点の距離とある基準となる長さの比率を利用する。比率を用いるのは、キャラクターの写り方でパーツの大きさに差があるからである。例として顔の横幅をそのままの長さで判別してしまうと、画像のサイズや写り方によって顔の横幅が違ってきってしまうからである。また、同じアングル同じ構図の画像があったとしても、画像の大きさが違うだけで顔の横幅に差が出てしまう。図 2.13 は、画像のサイズが違うだけで同じアングル同じ構図の画像である。顔の横幅である赤い線を比較すると画像のサイズが大きいものは顔の横幅が大きくなっている。また、たとえ同じ画像サイズであったとしても映っているキャラクターの大きさが異なっている場合も同様に顔の横幅が変わる。図 2.14 は、ある画像を縮小・トリミングしたものである。耳などの特徴点は髪の毛によって隠れる場合がある。顔の横幅を求めための特徴点は、特徴点の中で打ちやすい特徴点を利用して求めることができる。そのため、本研究では基準となる長さを顔の横幅と設定した。顔の横幅を構

成する特徴点も隠れる可能性もあるが、ある程度予測して特徴点を打つことができる。そのため、顔の横幅を構成する特徴点は、特徴点の中で打ちやすい点であると言える。

顔の縦幅と横幅の比率を求めるために、縦幅と横幅を求める。顔の横幅は、左右のこめかみの特徴点2点を結んだ線の長さを算出して求める。顔の縦幅を直接求めるためには、頭部の最上部に特徴点を打つ必要があるが、髪の毛に隠れてしまうのでユーザーが特徴点を指定するのが困難である。そのため本研究では、キャラクターイラストの描き方を参考にした [18]。キャラクターのイラストを描く場合、目の高さは、顔の縦方向において中心にくるように描く。これを利用して、顎から顔の中心までの長さを求め、その値を2倍にすることで間接的に顔の縦幅を求めることができる。顎から顔の中心までの長さは、顔の横幅の線に対して垂直で顎の特徴点を通る線の長さを利用する。図 2.15 における緑色の横線は顔の横幅を示し、緑色の縦線は顎から顔の中心までの長さを表し、この縦線を2倍した長さを顔の縦幅として利用する。よって、顔の横幅の長さ $l_{f,0}$ と縦幅の長さ $l_{f,1}$ を利用して、顔の縦幅と横幅の比率を求めることができる。キャラクター画像の顔の縦幅と横幅の比率になるように、モデルの顔の縦幅もしくは横幅に対して縮小を行う。顔のこめかみの左右の特徴点をそれぞれ $\mathbf{P}_{f,0}$ 、 $\mathbf{P}_{f,1}$ 、顎の特徴点を $\mathbf{P}_{f,2}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_{f,2}\mathbf{P}_{f,0}}$ のなす角を $\theta_{f,0}$ 、求める顔の横幅、縦幅をそれぞれ $l_{f,0}$ 、 $l_{f,1}$ とすると、 $l_{f,0}$ 、 $l_{f,1}$ はそれぞれ以下の式 (2.1) 及び式 (2.2) で求まる。

$$l_{f,0} = |\mathbf{P}_{f,1} - \mathbf{P}_{f,0}| \quad (2.1)$$

$$l_{f,1} = 2|(\mathbf{P}_{f,2} - \mathbf{P}_{f,0})\sin\theta_{f,0}| \quad (2.2)$$

したがって、求める比率 r_0 は以下の式 (2.3) で求まる。

$$r_0 = \frac{l_{f,1}}{l_{f,0}} \quad (2.3)$$

目の位置を求めるために、顔の横幅の線分と目の中心の最短距離を算出して利用する。目の中心は5つの特徴点の平均点を利用する。まず、右目の中心位置を求

め、その中心点と右のこめかみの特徴点のベクトルを求める。次に、そのベクトルの左右のこめかみの特徴点2点からなるベクトルへの正射影を利用する。よって、その射影ベクトルの長さとの顔の横幅の長さの比を求めることができる。同様に左側も求める。図 2.16 における緑色の線は顔の横幅を表し、青の点は左右それぞれの特徴点の中心を表し、桃色の線は目の位置を表している。なお、この図は見やすくするために顔の横幅である緑色の線は本来の位置より下にある。その後、キャラクター画像の目の位置関係になるように、モデルの目の位置を変更する。それぞれの目の特徴点を $\mathbf{P}_{e,i}(i=0,\dots,9)$ 右の特徴点を 0~4、左の特徴点を 5~9 とし、求める目の中心となる点を右目と左目をそれぞれを $\mathbf{P}_{e,r}$ 、 $\mathbf{P}_{e,l}$ とすると、 $\mathbf{P}_{e,r}$ 、 $\mathbf{P}_{e,l}$ はそれぞれ以下の式 (2.4) 及び式 (2.5) で求まる。

$$\mathbf{P}_{e,r} = \frac{1}{5} \sum_{i=0}^4 \mathbf{P}_{e,i} \quad (2.4)$$

$$\mathbf{P}_{e,l} = \frac{1}{5} \sum_{i=5}^9 \mathbf{P}_{e,i} \quad (2.5)$$

$\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_{e,r}\mathbf{P}_{f,0}}$ のなす角を $\theta_{e,0}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_{e,l}\mathbf{P}_{f,0}}$ のなす角を $\theta_{e,1}$ 、眉間から目の中心までの左右の長さをそれぞれ $l_{e,0}$ 、 $l_{e,1}$ とすると、 $l_{e,0}$ 、 $l_{e,1}$ はそれぞれ以下の式 (2.6) 及び式 (2.7) で求める。

$$l_{e,0} = |(\mathbf{P}_{e,r} - \mathbf{P}_{f,0}) \cos \theta_{e,0}| \quad (2.6)$$

$$l_{e,1} = |(\mathbf{P}_{e,l} - \mathbf{P}_{f,0}) \cos \theta_{e,1}| \quad (2.7)$$

したがって、求める比率 r_1 は以下の式 (2.8) で求まる。

$$r_1 = \frac{l_{e,0} + l_{e,1}}{2l_{f,0}} \quad (2.8)$$

目の比率は、顔の横幅に対する目の横幅の大きさを調べる。まず、右目の瞼の特徴点のうち中央の上瞼を除く2点を利用して右目の横幅を求める。この目の特徴点2点からなるベクトルのこめかみの特徴点2点からなるベクトルへの正射影を利用する。同様にして左目も行う。図 2.17 における緑色の線は顔の横幅を表し、赤い

線は目の長さを表す。その後キャラクター画像の目の大きさになるように、モデルの目の大きさを変更する。右目のこめかみに近い側の特徴点を $\mathbf{P}_{e,0}$ 、右目の眉間に近い側の特徴点を $\mathbf{P}_{e,1}$ 、左目のこめかみに近い側の特徴点を $\mathbf{P}_{e,5}$ 、左目の眉間に近い側の特徴点を $\mathbf{P}_{e,6}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_{e,0}\mathbf{P}_{e,1}}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_{e,5}\mathbf{P}_{e,6}}$ それぞれのなす角を $\theta_{e,2}$ 、 $\theta_{e,3}$ 、左右の目の幅をそれぞれ $l_{e,2}$ 、 $l_{e,3}$ とすると、 $l_{e,2}$ 、 $l_{e,3}$ はそれぞれ以下の式 (2.9) 及び式 (2.10) で求まる。

$$l_{e,2} = |(\mathbf{P}_{e,1} - \mathbf{P}_{e,0})\cos\theta_{e,2}| \quad (2.9)$$

$$l_{e,3} = |(\mathbf{P}_{e,6} - \mathbf{P}_{e,5})\cos\theta_{e,3}| \quad (2.10)$$

したがって、求める比率 r_2 は以下の式 (2.11) で求まる。

$$r_2 = \frac{l_{e,2} + l_{e,3}}{2l_{f,0}} \quad (2.11)$$

耳の位置は、顔の横幅の線分と耳の上端の特徴点の最短距離を求めて利用する。その後、顔の横幅の線分と、先程の最短距離の比を求める。図 2.18 における緑の線は顔の横幅を表し、青い線は横幅の直線と耳の上端の特徴点の最短距離を表す。その後、キャラクター画像の耳の位置になるように、モデルの耳の位置を変更する。右耳の上端にある特徴点を $\mathbf{P}_{a,0}$ 、左耳の上端にある特徴点を $\mathbf{P}_{a,2}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_{a,0}\mathbf{P}_{f,0}}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_{a,2}\mathbf{P}_{f,0}}$ それぞれのなす角を $\theta_{a,0}$ 、 $\theta_{a,1}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\mathbf{P}_{a,0}$ までの最短距離を $l_{a,0}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\mathbf{P}_{a,2}$ までの最短距離を $l_{a,1}$ とすると、 $l_{a,0}$ 、 $l_{a,1}$ はそれぞれ以下の式 (2.12) 及び式 (2.13) で求まる。

$$l_{a,0} = |(\mathbf{P}_{a,0} - \mathbf{P}_{f,0})\sin\theta_{a,0}| \quad (2.12)$$

$$l_{a,1} = |(\mathbf{P}_{a,2} - \mathbf{P}_{f,0})\sin\theta_{a,1}| \quad (2.13)$$

したがって、求める比率 r_3 は以下の式 (2.14) で求まる。

$$r_3 = \frac{l_{a,0} + l_{a,1}}{2l_{f,0}} \quad (2.14)$$

耳の大きさは、耳の上下の特徴点の距離を算出して利用する。そのため、まず耳の上下の特徴点 2 点のベクトルを求め、顔の横幅の線分に対して垂直な線を求める。

その線の長さを耳の大きさとする。図 2.19 における緑色の線は顔の横幅を表し、青い線は耳の大きさを表す。そうして求めた耳の長さ と 横幅の比率を計算する。その後、キャラクター画像の耳の大きさになるように、モデルの耳の大きさを変更する。右耳の下端にある特徴点を $\mathbf{P}_{a,1}$ 、左耳の下端にある特徴点を $\mathbf{P}_{a,3}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_{a,1}\mathbf{P}_{a,0}}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_{a,3}\mathbf{P}_{a,2}}$ それぞれのなす角を $\theta_{a,2}$ 、 $\theta_{a,3}$ 、 $\overrightarrow{\mathbf{P}_{a,1}\mathbf{P}_{a,0}}$ における $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ に対して垂直な成分を $l_{a,2}$ 、 $\overrightarrow{\mathbf{P}_{a,3}\mathbf{P}_{a,2}}$ における $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ に対して垂直な成分を $l_{a,3}$ とすると、 $l_{a,2}$ 、 $l_{a,3}$ はそれぞれ以下の式 (2.15) 及び式 (2.16) で求まる。

$$l_{a,2} = |(\mathbf{P}_{a,1} - \mathbf{P}_{a,0})\sin \theta_{a,2}| \quad (2.15)$$

$$l_{a,3} = |(\mathbf{P}_{a,3} - \mathbf{P}_{a,2})\sin \theta_{a,3}| \quad (2.16)$$

したがって、求める比率 r_4 は以下の式 (2.17) で求まる。

$$r_4 = \frac{l_{a,2} + l_{a,3}}{2l_{f,0}} \quad (2.17)$$

眉の位置は、顔の横幅の線分と眉の特徴点の最短距離を利用して算出する。こめかみから眉までのベクトルを求め、顔の横幅の線分に対して垂直な線の長さを求める。図 2.20 における緑色の線は顔の横幅を表し、青い線は横幅と眉の特徴点の距離を表す。これを利用して、顔の横幅の線分と眉の特徴点までの最短距離の比率を利用する。その後、キャラクター画像の眉の位置になるように、モデルの眉の位置を調整する。右眉の特徴点を $\mathbf{P}_{b,0}$ 、左眉の特徴点を $\mathbf{P}_{b,1}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_{b,0}\mathbf{P}_{f,0}}$ 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_{b,1}\mathbf{P}_{f,0}}$ それぞれのなす角を $\theta_{b,0}$ 、 $\theta_{b,1}$ 、 $\overrightarrow{\mathbf{P}_{b,0}\mathbf{P}_{f,0}}$ における $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ に対して垂直な成分を $l_{b,0}$ 、 $\overrightarrow{\mathbf{P}_{b,1}\mathbf{P}_{f,0}}$ における $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ に対して垂直な成分を $l_{b,1}$ とすると、 $l_{b,0}$ 、 $l_{b,1}$ はそれぞれ以下の式 (2.18) 及び式 (2.19) で求まる。

$$l_{b,0} = |(\mathbf{P}_{b,0} - \mathbf{P}_{f,0})\sin \theta_{b,0}| \quad (2.18)$$

$$l_{b,1} = |(\mathbf{P}_{b,1} - \mathbf{P}_{f,0})\sin \theta_{b,1}| \quad (2.19)$$

したがって、求める比率 r_5 は以下の式 (2.20) で求まる。

$$r_5 = \frac{l_{b,0} + l_{b,1}}{2l_{f,0}} \quad (2.20)$$

鼻の位置は、顔の縦幅に対する位置関係を求める。そのために、顔の幅の線分と鼻の特徴点の最短距離を求める。図 2.21 における緑色の線は顔の横幅を表し、青い線は横幅と鼻の特徴点の距離を表す。その後、キャラクター画像の眉の位置になるように、モデルの鼻の位置を調整する。鼻の特徴点を \mathbf{P}_n 、 \mathbf{P}_n を通り $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ に垂直な線と $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ の交点を \mathbf{P}_c 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_n\mathbf{P}_{f,0}}$ のなす角を θ_n 、鼻までの高さを l_n とすると以下の式 (2.21) で求まる。

$$l_n = |(\mathbf{P}_n - \mathbf{P}_{f,0})\sin \theta_n| \quad (2.21)$$

したがって、求める比率 r_6 は以下の式 (2.22) で求まる。

$$r_6 = \frac{l_n}{l_{f,0}} \quad (2.22)$$

口の位置は、顔の縦幅に対する位置関係を求める。そのために、顔の幅の線分と口の特徴点の最短距離を求める。図 2.22 における緑色の線は顔の横幅を表し、青い線は横幅と口の特徴点の距離を表す。その後、キャラクター画像の眉の位置になるように、モデルの鼻の位置を調整する。口の特徴点を \mathbf{P}_m 、 \mathbf{P}_m を通り $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ に垂直な線と $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ の交点を \mathbf{P}_c 、 $\overrightarrow{\mathbf{P}_{f,1}\mathbf{P}_{f,0}}$ と $\overrightarrow{\mathbf{P}_m\mathbf{P}_{f,0}}$ のなす角を θ_m 、口までの高さを l_m とすると以下の式 (2.23) で求まる。

$$l_m = |(\mathbf{P}_m - \mathbf{P}_{f,0})\sin \theta_m| \quad (2.23)$$

したがって、求める比率 r_7 は以下の式 (2.24) で求まる。

$$r_7 = \frac{l_m}{l_{f,0}} \quad (2.24)$$



図 2.13: 画像の大きさの違い



図 2.14: 写り方の違い

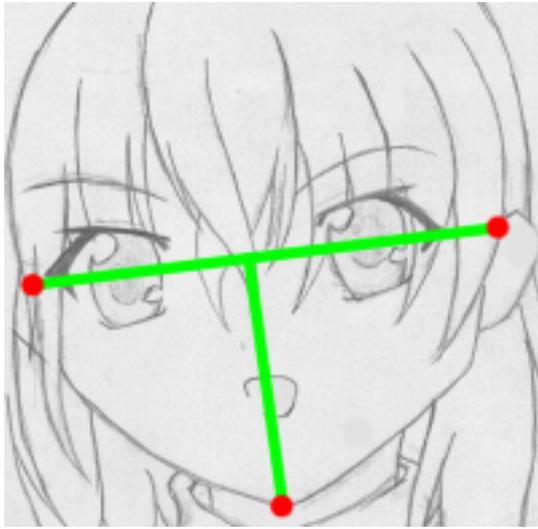


図 2.15: 顔の縦幅と横幅の比率の分析

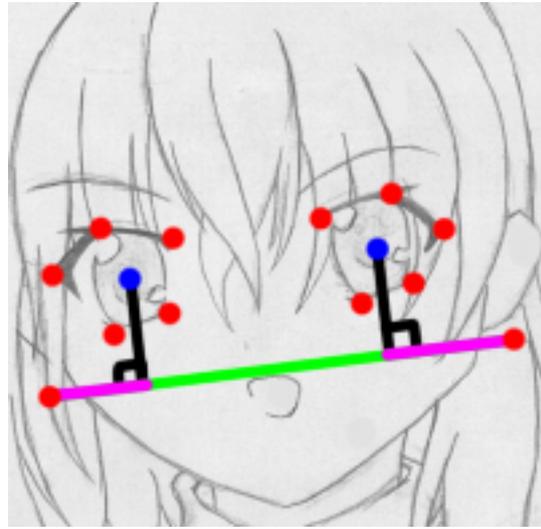


図 2.16: 目の位置の分析

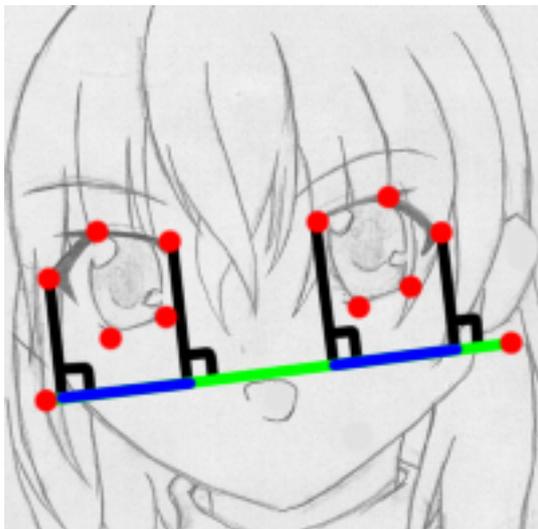


図 2.17: 目の比率の分析

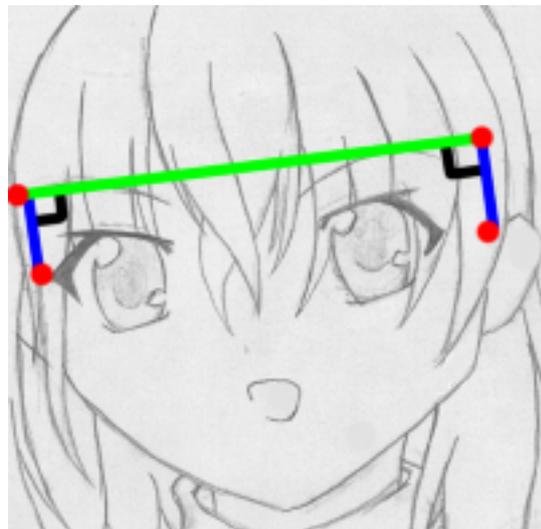


図 2.18: 耳の位置の分析

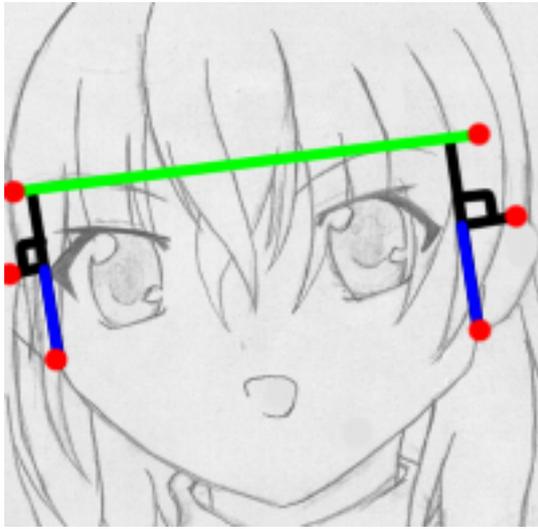


図 2.19: 耳の大きさの分析

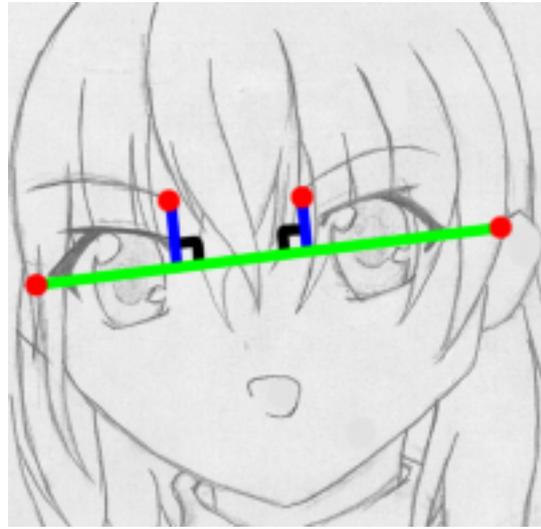


図 2.20: 眉の位置の分析

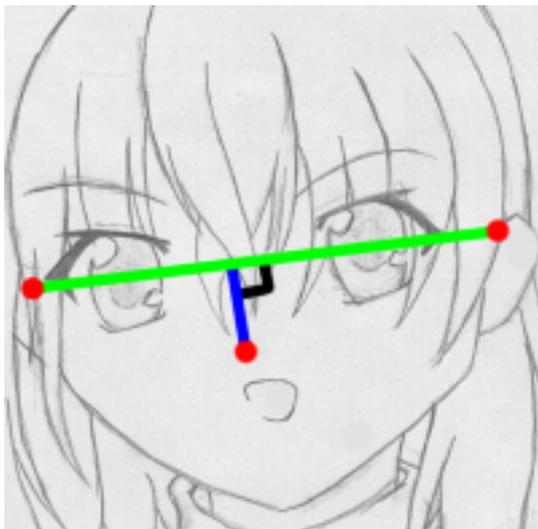


図 2.21: 鼻の位置の分析

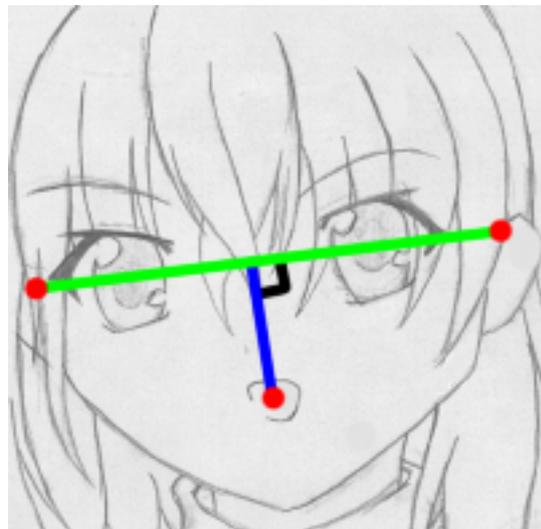


図 2.22: 口の位置の分析

第 3 章

システムの検証と考察

本章では、第 2 章で提案した手法を検証する。まず、どのような方法を用いて検証を行ったかを述べる。そのために、第 2 章のシステムを実装する必要があるため、実装について述べる。最後に、実装したシステムを利用した検証の結果から本手法の有効性を考察する。

3.1 検証方法

本研究の有用性を確かめるために、人がパラメータを調整する従来の方法と特徴点を利用する本手法を比較する。まず、従来の手法と本手法それぞれを利用してパラメータ調節を行ったモデルにおいて、似ている形状である必要がある。この場合の似ているとは、従来の手法と本手法のパラメータの数値が近いというのではなく、人が見たときの印象を指す。そのために、予め両手法でパラメータ調節を行ったモデルを比較してもらい、よく似ている、どちらかというとき似ている、どちらかというとき似ていない、似ていないの 4 段階で判断してもらった。次に、もととなった画像と両者のモデルを比較してもらい、どちらが似ているかを判断してもらった。評価は、先程の検証と同様によく似ている、どちらかというとき似ている、どちらかというとき似ていない、似ていないの 4 段階とした。本手法では髪の毛に関する変形を考慮していないが、髪の毛は人に与える印象が強いので

で[19]、髪の毛のあるものとなないものの比較も行った。髪の毛はもととなったキャラクター画像の髪の毛の部分を切り取って表示させ表現した。最後に、それぞれの手法でパラメータ調整を終えるまでの所要時間を比較する。人がパラメータ調節を行う場合、パラメータの調節を始めた時間から全てのパラメータの調節を終えた時間を利用する。本手法の場合、予めキャラクター画像を読み込んだ状態からはじめ、特徴点の打ち始めから時間を計測し、特徴点を打ち終わるまでの時間を利用する。なお、被験者には予め本手法と従来の手法それぞれのシステムを触れてもらい、ある程度操作する感覚を掴んでから時間の計測を行った。

3.2 実装

本手法を検証するためにプログラムの作成を行った。開発言語はC++、開発環境はMicrosoft Visual Studio 2008、ライブラリとしてOpenGL[20]をベースとしたFine Kernel ToolKit[21]を利用した。機能として画像の読み込み、画像に対して特徴点を指定、パラメータの調節を行うことでリアルタイムに形状変形を行うことができるものを作成した。

3.3 結果と考察

今回19人に検証実験ののちアンケートを実施した。その結果をそれぞれ示す。今回は被験者に対して5種類のモデルを比較してもらった。次の表3.1、表3.2は2つのモデルを比較して似ているかの調査結果であり、表3.3、表3.4、表3.5、表3.6は元になった画像と比較してどちらが似ているかの調査結果である。それぞれの評価は、4を「よく似ている」、3を「どちらかというとき似ている」、2を「どちらかというとき似ていない」、1を「似ていない」の4段階である。また、パラメータ調整するまでの所要時間を比較する。表3.7はその結果である

今回の結果として、人が作成したものとパラメータ調節を行った場合では見た目の印象に大きな差がなかったと言える。そして、作業時間に関しても大きな差

がなかった。しかし、意見の中には特徴点を利用したほうが早く感じられたという意見があった。また、パラメータを調節する従来の手法と比較して、画像に対して点を打ってあげればいいので簡単で良いという意見がでた。これは、絵の上に点を配置するだけなので直感的に操作ができるためである。このような意見はキャラクターメイキングをあまり行ったことのない人や大体の形状が出来れば良いという人に多い傾向があった。そして、大きな変形やそれによるバランス調整を考える必要がないので簡単だという意見があった。

今回の結果やシステムを触ってもらった感想や意見からわかった問題点を挙げる。主な問題点として、変形箇所不足、ユーザーインターフェース、キャラクター画像の3つがある。

第1の問題は、本研究ではパラメータを利用したキャラクターメイキングの変形できる部位が少なかったことである。実際のキャラクターメイキングでは本研究で利用した部位以外にも、目や眉のツリ具合といった傾き、鼻の高さといったものを変えることができる。こう言った部位の変化は印象に大きく関わってくる。キャラクターの目の傾きを例にあげると、つり目であるキャラクターは印象が凛々しく、タレ目では大人しく感じる。今回利用した部位だけでは、そういったものを再現することができなかつたため、作成されたキャラクターが似ていないと感じる傾向にあった。意見の中にも、目のツリ具合などでキャラクターの形状を判断するので必要だという意見もあった。また、パラメータの調節を利用して変形を行わないということで、本手法ではヘアスタイルの調整は行わないとしていた。しかし、表3.1と表3.2表3.4と表3.6、表3.3と表3.5をそれぞれ比較するとわかるように、髪の毛があるというだけで似ていると感じやすかった。キャラクターの印象を近づけるためには、ヘアスタイルの自動調節を行う必要があると言える。そのため、本研究ではパーツを利用して行うヘアスタイルの選択、色の調整を行うことは考慮しなかったが、これらの調節するシステムを追加する必要がある。

第2の問題は、ユーザーインターフェースの問題である。今回は特徴点に数字を書き、その点がどこに対応するかという表を見せて特徴点を打っていた。しか

し、数字ではどこに対応するかということが直感的に理解出来ない。また、従来の手法の検証用のプログラムではパラメータ調節領域とモデルの表示領域があった。そのためユーザはパラメータ調節を行った時リアルタイムに変化の様子が分かった。だが、本手法のプログラムではキャラクター画像に特徴点を表示する領域しか設けなかった。なので、変形の様子がリアルタイムに表示できていなかった。そのため、従来の手法ではリアルタイムな変化が分かり、本手法ではわからなかったため、従来の手法が使いやすいという意見が出た。特徴点に関しては、点選択時、表示領域の何処かにどの部位におくかわかるガイドを設けることで解消することができる。リアルタイム性は、従来の手法と同様にモデルを表示しておき変化の様子がわかるようにする必要がある。

最後の問題は、キャラクター画像における顔の移り方や表情の問題である。画像によっては髪の毛などによって特徴点を打つ場所が隠れている場合、特徴点の位置が判断しづらくなってしまう。主な部位として、耳の位置やこめかみの位置は髪の毛の影響を受け易い。ある程度予測をつけて打つことはできるが、キャラクターイラストの描き方や人の骨格構造を知っている必要があるので、ユーザーへの負担がかかってしまう。部位が隠れている場合はキャラクターイラストの描き方を参考にして耳や顔の輪郭を描いてくれる補助システムがあれば、部位が隠れていてもユーザは特徴点を打ちやすくなる。

3.4 今後の展望

3.3節のアンケートで変形部位や色についての意見が多かった。そのため、変形部位の数を増やすこと、髪の毛の選択を可能にする、色を自動的に設定することが重要であることがわかる。髪の毛や色を自動的に調整することでより理想的なキャラクターに近づけることが可能になる。

次に、ユーザーインターフェースでの問題は重要であると言える。ユーザーインターフェイスがよくなればユーザーへ掛かる負担は減る。そのために、ユーザーインターフェイスはモデルの変化がリアルタイムにわかるようにモデルを表示す

る領域と画像に特徴点を打つ領域をそれぞれ設ける。また、特徴点の対応関係をひと目で見て理解できるように、点を選択したら操作画面の何処かに点の具体的な位置を示す必要がある。

最後に、本研究で提案した手法では、特徴点を指定する手間がかかるという問題がある。ユーザーにはパラメータを調節する作業ではなく点を打つ作業を課している為、ユーザーへの負担に大きな差がない。こういった問題の解消方法として、OpenCV[22] のオブジェクト検出を利用が考えられる。これを利用し、キャラクターの顔の検出を行い、検出されたキャラクターの顔に対して特徴点を自動で取得する、もしくは特徴を自動的に判断することができれば、ユーザーへの負担を減らすことが可能になる。

表 3.1: 髪の毛のない状態の2つのモデルの比較

	ケース0	ケース1	ケース2	ケース3	ケース4
平均	3.0	2.3	2.1	3.9	3.4

表 3.2: 髪の毛のある状態の2つのモデルの比較

	ケース0	ケース1	ケース2	ケース3	ケース4
平均	2.8	3.0	2.1	3.7	3.5

表 3.3: 元の画像と本手法で作成した髪の毛のないモデルの比較

	ケース0	ケース1	ケース2	ケース3	ケース4
平均	1.5	2.1	1.9	1.9	2.2

表 3.4: 元の画像と従来の手法で作成した髪の毛のないモデルの比較

	ケース0	ケース1	ケース2	ケース3	ケース4
平均	2.3	1.7	2.1	1.7	2.3

表 3.5: 元の画像と本手法で作成した髪の毛のあるモデルの比較

	ケース0	ケース1	ケース2	ケース3	ケース4
平均	1.8	2.8	2.2	2.3	2.5

表 3.6: 元の画像と従来の手法で作成した髪の毛のあるモデルの比較

	ケース0	ケース1	ケース2	ケース3	ケース4
平均	2.6	2.6	2.5	2.2	2.5

表 3.7: 所要時間 (秒)

	本手法	パラメータ
平均	110	102

第 4 章

まとめ

本研究では、パラメータを利用したキャラクターメイキングにおいてパラメータ調節にかかる時間を短縮する手法について提案、検証を行った。本手法は、ユーザーが特徴点を直接指定して、それらの特徴点を利用してパーツの位置関係や大きさを算出し、そのデータをもとにしてパラメータを自動で調節するものである。3章の検証から、あまりいい結果を得ることができなかつた。主な要因としては、髪の毛の調整、色の調整、変形箇所が不足が挙げられる。現段階では、顔パーツごとの位置関係や大きさといったものなど限られたものに対してしか行えていない。キャラクターの印象を変えることが出来る部位に対してのパラメータ調節や、ヘアスタイルの変更ができれば、より実用的になるだろう。

また、ユーザーインターフェースを考えずシステム構成を行ってしまった。特徴点の配置箇所やリアルタイムにモデルの変形がわからないという問題があった。こういった点への配慮をした設計にしておすことで、ユーザーへの負担は軽減されるだろう。

しかし、従来の手法に慣れている人から、従来の手法では1からパラメータを設定するので、ベースとなるパラメータがすぐに設定されているとやりやすいという意見があった。現段階では、本手法をそのまま利用するよりも従来の手法のみを利用してキャラクターメイキングを行うと早い。それは、本手法ではユーザーに特徴点を指定するという新たな作業を課している為である。この点を打つ作業

の自動化、もしくは直接特徴を自動的に判定することができれば、ユーザーへの負担が減る。そうなれば、本手法の有用性が高まるはずである。

謝辞

本研究を進めるに当たり、渡辺大地講師、三上浩司講師、同研究室の院生および学部生の方々には大変お世話になりました。心から感謝致します。

また、研究の半ば頃、精神的にまいってしまいお見苦しい姿を見せてしまいました。そのことで気分を害してしまった方が居らっしゃると思いますので、この場を借りてお詫びいたします。

そして、アンケートに協力してくれた皆様、大変ありがとうございます。厳しい意見や私では気づかなかった点など大変参考になりました。

参考文献

- [1] 株式会社インセプト, “アバターとは【avatar】 - 意味/解説/説明/定義: IT用語辞典.” <http://e-words.jp/w/E382A2E38390E382BFE383BC.html>.
- [2] 株式会社アルファ・システム, 株式会社セガ, “ファンタシースターポータブル 2,” (12 2009).
- [3] MGAME JAPAN Corp., “ARGO ONLINE,” (1 2011).
- [4] 北京完美時空ネットワーク技術有限公司, “パーフェクトワールド-完美世界-,” (4 2007).
- [5] エヌ・シー・ソフト, エヌ・シー・ジャパン, “The tower of aion,” (7 2009).
- [6] 梁良, 石亀 昌明, 小嶋 和徳, 伊藤 慶明, “配置と個人特徴の誇張を利用した似顔絵作成システム,” **第 8 回情報科学技術フォーラム** (2007).
- [7] 武本 征士, 高木 佐恵子, 吉本 富士市, “色も誇張された 3 次元似顔絵の生成,” **第 20 回 NICOGRAPH 論文コンテスト 論文集** (2004).
- [8] Takeo Igarashi, Satoshi Matsuoka, Hidehiko Tanaka, “Teddy A Sketching Interface for 3D Freeform Design,” *SIGGRAPH 99* (1999).
- [9] Olga Karpenko, John F. Hughes, Ramesh Raskar, “Free-from Sketching with Variational Implicit Surfaces,” *Computer Graphics Forum* (2002).

- [10] 吉田 武史, 高井 那美, 高井 昌彰, “セルアニメーションを用いたキャラクター頭部 3D モデル生成支援,” **情報処理学会研究報告** (2005).
- [11] 小野 豊, 陳 炳宇, 西田 友是, “セルアニメーションからの 3 次元キャラクタモデルの生成とその応用,” **情報処理学会研究報告** (2003).
- [12] 森 大樹, 箕浦 大祐, 前田 泰宏, 安野 貴之, “3D デフォルメキャラクタの自動生成における顔特徴量の獲得およびモデルへの反映,” **電子情報通信学会総合大会講演論文集** (2001).
- [13] 森 大樹, 本田 新九郎, 安野 貴之, 石橋 聡, “人物顔形状を反映した 3D デフォルメキャラクタ生成システムの構築,” **電子情報通信学会技術研究報告** (2001).
- [14] 森 大樹, 箕浦 大祐, 前田 泰宏, 本田 新九郎, 安野 貴之, 石橋 聡, “3D デフォルメキャラクタ生成システムにおける顔 3 次元モデリング方法の検討,” **電子情報通信学会技術研究報告** (2001).
- [15] 今尾 公二, 亀井 能成, 美濃 導彦, 池田 克夫, “シルエット画像に基づいて個人体系を反映する 3 次元形状モデルの変形方法-仮想試着室の実現に向けて,” **知能情報メディア論文集** (1999).
- [16] 森 康紘, “固有顔ベクトルを使った 3 D 顔モデルの研究,” **九州大学修士論文** (2008).
- [17] 株式会社フロム・ソフトウェア, 株式会社ソニー・コンピュータエンタテインメント, “Demon’s souls,” (2 2009).
- [18] 林 晃, [スーパーマンガデッサン—作画のための考えるデッサン], グラフィック社 (2005).
- [19] 金子智栄子, 門脇幹雄, “外見の印象 -髪型が性格のイメージに及ぼす影響-,” **文京学院大学研究紀要** (2001).

- [20] Silicon Graphics International., “OpenGL.” <http://www.opengl.org/>.
- [21] 渡辺 大地, “Fine Kernel Tool Kit System.” <http://fktoolkit.sourceforge.jp/>.
- [22] Intel, “OpenCV.” <http://opencv.willowgarage.com/>.