

モデルの連続変化映像の推薦によるキャラクターメイキング手法 「SteerMorph」のアバター作成に関する評価

竹永正輝¹ 橋本直¹

概要：ゲーム内でカスタマイズ可能なアバターを作成する機能として、キャラクターメイキングがある。既存 UI では作成できるアバターのバリエーションの増加に伴ってデザインにかかる作業負担も増加する。そこで本研究では、モデルの連続変化映像の推薦からキャラクターメイキングを行う手法「SteerMorph」を提案する。提案手法では、StyleGAN によって生成されたアバター画像をアニメーションとして提示する。ユーザは提示画像に対して気に入った度合いを入力する。システムは一定の評価データ蓄積後に対話型進化計算を行い、ユーザの好みを反映させた出力画像を提示する。これらの処理は繰り返し行われ、ユーザはアニメーションを見ながら1軸のスティック操作を行うだけでアバターを作成できる。本稿では、顔画像を対象とした実装を行い、評価実験を行った結果について報告する。

キーワード：キャラクターメイキング, 機械学習, 対話型進化計算

1. はじめに

オンラインゲームや Web コンテンツ内でユーザがカスタマイズ可能なアバターを作成する機能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の UI として、顔や胴体のパーツを付け替えるものやそれらの色やサイズをスライダー操作で変更するものがある。このような UI では、パーツの種類やパラメータの数が多いほど、作成できるアバターのバリエーションは多くなるが、自由度の増大に伴ってユーザの作業負担は増大し、思い通りのデザインにすることが難しくなる。また、プリセットを改良する方法もあるが、さまざまなユーザの希望に対応するためには多くのプリセットを用意する必要がある。プリセットの数が多くなるほど改良にかかる作業負担は減る一方で、多くの候補の中から気に入ったものを選択する必要がある。この他にもランダム生成された候補から選択する方法もあるが、思い通りのデザインが出力されるかどうかは運に左右されるため、満足する結果をすぐに得られることもあれば、そうでないときもある。

そこで本研究では、キャラクターメイキングにおける作業負担の軽減や効率向上を目的として、モデルの連続変化映像の推薦からキャラクターメイキングを行う手法「SteerMorph」を提案する。提案手法では、StyleGAN[1]によって生成されたアバター画像をアニメーションとして提示する。ユーザは提示される画像に対して、気に入った度合いを評価する。システムは一定の評価データ蓄積後に対話型進化計算[2]を行うことで、ユーザの好みを反映させた出力画像を提示する。これらの処理は繰り返し行われ、ユーザはアニメーションを見ながら1軸のスティック操作を行うだけでアバターを作成できる。

一般的なキャラクターメイキングにおいて顔の部位は最も自由度が高く、時間がかかりやすい箇所であるため、

本研究では顔を対象としたキャラクターメイキングを行った。本稿では、顔を対象としたキャラクターメイキングを行い、評価実験を行った結果について報告する。

2. 関連研究

アバター作成を目的に機械学習を利用することで、大量の顔画像を用いた学習データを使用して多様な表現力を持った画像生成を行うことができる。またユーザの主観を入力として扱い、画像の出力に反映させることでインタラクティブな画像生成を行う手法はさまざまなものが提案されている。Philip ら[3]は複数の潜在変数のベクトルを元に生成した顔画像に対話型進化計算を行うことでモデルの進化を行った。これにより大量の画像を用いたデータセットを事前に用意することで、顔画像をユーザの入力によって特定のテーマに基づいた進化をさせることが可能になった。MakeGirls.moe[4]は画像生成に用いるノイズを調節することで、より希望に沿った顔画像を生成することができる Web サービスである。クライアント側で計算を行うことで、2 画像間のモーフィング画像の出力も行うことができる。WaifuLabs[5]は基準となるモデルを選択した後、16 種類の顔画像の候補から気に入ったものを選択することで徐々に特定要素を変化させていくことができる Web サービスである。配色や絵柄、ポーズといった特定要素の学習がなされており、基準となる顔画像は特定要素を用いて進化させていくことができる。Crypko[6]は顔画像の生成を行い、それを販売できる Web サービスである。ユーザは顔画像の生成を行い、その画像の質を判断する。質の高いデータを購入することもでき、そのデータを用いることでさらに質の高い画像生成を行える。この取引を繰り返すことでユーザはより質の高い画像生成を行う。

我々の提案手法では、WaifuLabs と同様にユーザの入力に合わせて顔画像を変化させるが、ユーザの作業負担を軽

¹ 明治大学
Meiji University

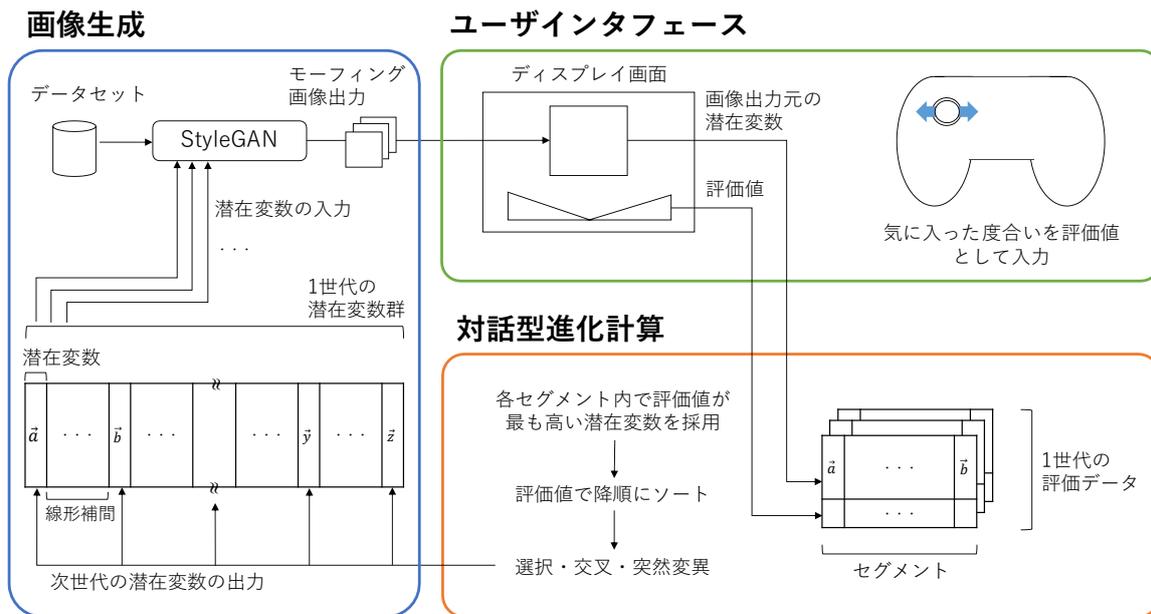


図1 概要図

減させるために1軸のスティック操作を使用するアプローチをとった。

3. 提案手法

SteerMorphは、生成したモーフィング画像を短時間で更新することでアニメーションとして提示し、それに対する評価データをもとに対話型進化計算を行うキャラクターメイキング手法である。本稿では主にゲーム内で用いる顔テクスチャやWebコンテンツの顔アイコンなどで用いる画像生成を目的とした実装方法を説明する。

提案手法の概要を図1に示す。また提案手法のフローチャートを図2に示す。提案手法は画像生成、対話型進化計算、ユーザインタフェースから構成される。構成要素のそれぞれについて説明する。

3.1 画像生成

StyleGANはGAN (Generative Adversarial Networks) [7]におけるアーキテクチャの一種である。GANのgenerator層にスタイル変換の技術を応用することで、高品質な画像生成が行える。StyleGANは512次元の配列で構成された乱数ベクトルを入力することで、画像出力を行う。また2つのベクトルの平均ベクトルを用いることで中間画像の生成が行える。本手法で用いるモーフィング画像の生成のために必要な、乱数ベクトルの生成アルゴリズムを説明する。本アルゴリズムは、一定の区間に分けて画像の補間を行う。まず開始と終了に用いる乱数ベクトルを生成する。1画像は短時間表示され、複数フレームをかけてベクトルの各値を線形補間する。これを1セグメントとする。次のセグメントに移る際、前セグメントの終了ベクトルを開始ベクトルとし、終了ベクトルを新たに生成する。同様に線形補間

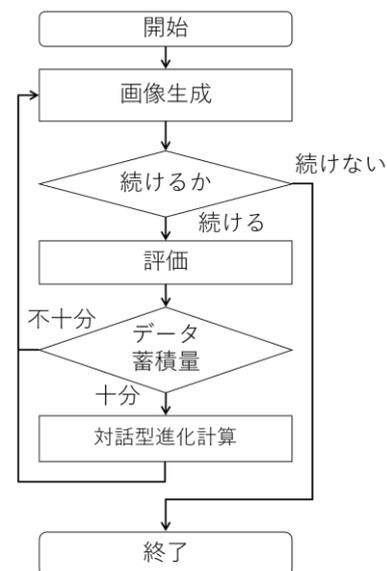


図2 フローチャート

を行うことで、新たにセグメントの生成を行う。こうして作成されたベクトルを用いてStyleGANで画像を生成することで、モーフィング画像を出力する。

3.2 対話型進化計算

一定のデータを蓄積した後、対話型進化計算を行う。対話型進化計算を行う際は1セグメントの中で最も評価の高い潜在変数を代表値として用い、代表値の集団で評価値を降順にソートすることでランキング選択を行う。

対話型進化計算には、遺伝的アルゴリズム (Genetic Algorithm: GA) を用いた。GAは生物の進化メカニズムを模倣しており、選択、交叉、突然変異を繰り返すことでより適応度の高い個体を生成することができる。対話型進化計算によるモデルの収束例を図3に示す。本手法は親の選

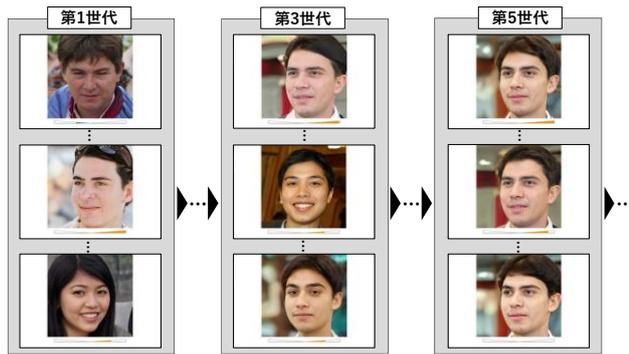


図3 対話型進化計算によるモデルの収束



図5 ランダム法



図4 ユーザインタフェース

択法にランキング選択を用いて、一意な組み合わせの遺伝子を2つ選択する。交叉には割合で交叉させる方式をとった。これは親の遺伝子としての特徴を保ちつつ、子の生成を行うためである。交叉に用いる数式を以下に示す。

$$\vec{c} = r \cdot \vec{a} + (1-r) \cdot \vec{b}$$

\vec{a} , \vec{b} , \vec{c} はベクトルとして表現される潜在変数であり、 r は配合率 ($0 \leq r \leq 1$) である。突然変異の方法として、摂動と交換を行う。摂動ではランダムな遺伝子座に数値加算を行う。

3.3 ユーザインタフェース

ユーザインタフェースを図4に示す。キャラクターメイキングのUIとして、画面中央にStyleGANにより生成された顔画像が表示され、その下に画像評価のためのゲージが表示される。ユーザはPCに接続されたゲームコントローラのスティック入力を用いて画像の評価を行う。

ユーザがコントローラの開始ボタンを押すと、画面には性別や年齢、髪型や肌の色などが異なるさまざまな顔画像がモーフィングしながら提示される。ユーザはそれを見ながらコントローラのアナログスティックを左右に傾けることで評価を行う。表示された画像が気に入ったときは右側に、そうでないときは左側に傾ける。また、その度合いに応じた角度でスティックを傾ける。この評価値に応じて画面下部のゲージが増減する。提示される画像はユーザが評価を行うことで徐々に収束する。ユーザは停止ボタンを押

すことで、顔画像の連続変化を止められる。満足した画像が作られたときには終了ボタンを押して画像を保存し、キャラクターメイキングを終了する。ユーザはリセットボタンを押すことで望まない収束を迎えたときや、気に入らないと感じたときにははじめからやり直すことができる。

3.4 実装

提案手法を、Pythonを用いて実装した。画像生成にStyleGAN[a]を、顔画像データセットにはFlickr-Faces-HQ Dataset[b]を用いた。コントローラの入力と画面描画にはPygame[c]を用いた。

4. 実験

キャラクターメイキングにおける作業負担の軽減やその効率向上を目的に評価実験を行った。評価実験は提案手法であるSteerMorphに加え、ランダム法(ランダム生成手法)、進化計算法(対話型進化計算手法)でキャラクターメイキングを行い、アンケートを実施した。

4.1 実験環境

研究室内において、参加者を着席させた状態で実験を行った。PCはRazer blade 15 (CPU: Intel Core i7-9750H, GPU: GeForce RTX 2060, RAM: 16GB)を使用した。コントローラはXboxコントローラを使用した。実験者は操作説明を行うために参加者の付近で待機した。

4.2 実験条件

キャラクターメイキングの経験がある情報系の大学生10名(21~22歳, 男性7名, 女性3名)を対象に実験を行った。比較実験のために作成したランダム法, 進化計算法について説明する。

(1) ランダム法

ランダム法はランダムな疑似乱数列を用いて、気に入った画像が現れるまで生成を行う方式である。ランダム法の画面を図5に示す。ディスプレイにはStyleGANによって生成された顔画像が6枚表示されており、3枚ずつ2段に

a) "StyleGAN". <https://github.com/NVlabs/stylegan> (参照 2020-2-21).

b) "Flickr-Faces-HQ Dataset". <https://github.com/NVlabs/ffhq-dataset> (参照 2020-2-21).

c) "Pygame". <https://www.pygame.org> (参照 2020-2-21).



図6 進化計算法

分けて表示されている。Nishino ら[8]の研究によると、モバイル端末においてモデルの表示数を多くすると視認性が下がる一方で、モデルの作成にかかる時間が短くなる傾向があることが報告された。本実験は PC に接続されたディスプレイに顔画像の表示を行うが、それぞれの顔の造形を確認することを考慮して、6 枚画像提示を行った。ユーザが選択中の画像は周りが青く表示される。選択移動ボタンによって画像の選択を左右に切り替えることができ、画面右下の画像を選択しているときに生成ボタンを押すことで、新たに 6 枚画像が生成される。なお、右上画像の次の選択は左下画像に対応している。満足した画像が作られたときには終了ボタンを押すことで選択中の画像を保存し、キャラクターメイキングを終了する。

(2) 進化計算法

進化計算法はランダム法と対話型進化計算を組み合わせた手法である。進化計算法の画面を図 6 に示す。ランダム法と同様にディスプレイには StyleGAN によって生成された顔画像が 6 枚表示されており、3 枚ずつ 2 段に分けて表示されている。顔画像の下には評価用のボタンが 5 つ表示されていて、選択中の評価ボタンは灰色になっている。画面レイアウトには西野ら[9]の研究を参考にした。ユーザが選択中の画像は周りが青く表示される。ユーザは表示された画像に対して 5 段階評価を行い、SteerMorph と同様の対話型進化計算を行う。与えられた評価は降順にソートされ、ランキング選択を行うが、同評価内はデータ蓄積順で順位付けが行われるという問題がある。そのため評価データ蓄積後に全体のデータ順序のシャッフルを行い、蓄積順序による対話型進化計算を行う際の選択影響をなくすようにした。ユーザの操作はランダム法と同様に選択移動ボタンによって画像の選択を左右に切り替えることができ、画面右下の画像を選択しているときに生成ボタンを押すことで、新たに 6 枚画像が生成される。なお、右上画像の次の選択は左下画像に対応する。また評価変更ボタンによって

選択中の画像の評価を変更することができる。満足した画像が作られたときには終了ボタンを押して画像を保存し、キャラクターメイキングを終了する。ユーザはリセットボタンを押すことで望まない収束を迎えたときや、気に入らないと感じたときにははじめからやり直すことができる。

4.3 設定パラメータ

SteerMorph の画像生成パラメータはなめらかなアニメーション提示を行うために各画像の表示時間を 0.25 秒、補間数を 10、セグメント数を 12 とした。進化計算法の親の候補数は 12 とした。対話型進化計算のパラメータは交叉確率を 95%、配合率を 40%、変異確率は摂動と交換のそれぞれを 10% とした。摂動の範囲は -1.0~1.0 とした。またランキング選択の割合には 2 順位ごとに確率が半減されていく重みづけを用いた。ウィンドウの大きさは横幅を 1200 ピクセル、縦幅を 800 ピクセルとした。

4.4 実験手順

実験は参加者内比較とし、全参加者に 3 種類すべての条件を体験させた。実験順はカウンターバランスをとった。実験前にキャラクターメイキングの経験に関するアンケートを実施した。その後キャラクターメイキングのAvatarで表現する世界観の共有を図るために、参加者に既存ゲームの予告編映像[d][e]の一部を見せ、この風景で生活する人物を作成させた。なお使用したデータセットの都合上、サングラスやイヤリングといった装飾物のついた画像が生成されることもあるが、顔の特徴に着目してキャラクターメイキングを行うように指示した。各手法は操作説明を行った後にキャラクターメイキングを行い、タスク達成時間と成果画像を記録した。そして条件ごとに手法に関するアンケートを実施した。アンケートは 5 段階のリッカート尺度を用いた。その項目を表 1 に示す。また、実験終了後に聞き取り調査を行った。

4.5 結果

各手法においてタスク達成時間と体感による時間、およびリセット数の平均を表 2 に示す。体感による時間よりもタスク達成時間はいずれも短くなった。またタスク達成時間が長くなるほど、体感による時間との差が大きくなる傾向がみられた。実験前アンケートから得られた回答では普段キャラクターメイキングにかかる時間の平均は 21.7 分であり、通常よりもいずれも短くなった。本実験で得られたタスク達成時間から作業効率の向上は果たせなかったが、SteerMorph、進化計算法では手法のリセットが数回行われていた。このことから初期集団の影響や望まない画像の収束が行われたことが考えられる。それらの影響を加味して、手法ごとに 1 回の画像作成にかかるタスク達成時間を導出する。1 回のタスク達成時間の算出式を以下に示す。

d) “Watch Dogs 2 Trailer: Cinematic Reveal - E3 2016 | Ubisoft [NA]” . <https://www.youtube.com/watch?v=hh9x4NqW0Dw> (参照 2020-2-21).
e) “Watch Dogs 2 - Launch Trailer | Ubisoft [NA]” . <https://www.youtube.com/watch?v=2GIVVsTKLTg> (参照 2020-2-21).

表1 実験後アンケートの質問項目

Q1	体感でどのくらいの時間キャラメイクを行いましたか (分)
Q2	操作が簡単だった (1: そう思わない~5: そう思う)
Q3	キャラメイクを行うことが楽しかった (1: そう思わない~5: そう思う)
Q4	操作が面倒だと感じた (1: そう思わない~5: そう思う)
Q5	この方法でキャラメイクを行うのは疲れた (1: そう思わない~5: そう思う)
Q6	自分の入力によって、より自分の希望に近いものが表示されたと感じた (1: そう思わない~5: そう思う)
Q7	満足したものが作れた (1: そう思わない~5: そう思う)
Q8	他のゲームでもこの方法でキャラメイクを行いたいと思う (1: そう思わない~5: そう思う)
Q9	この方法はキャラメイクに適切だと感じた (1: そう思わない~5: そう思う)
Q10	気づいたことがあれば自由に書いてください (自由記述)

表2 手法ごとの時間とリセット数の平均

手法	タスク達成時間 (分)	体感による時間 (分)	リセット数
SteerMorph	4.90	6.90	1.9
進化計算法	4.08	5.25	0.9
ランダム法	1.92	2.85	

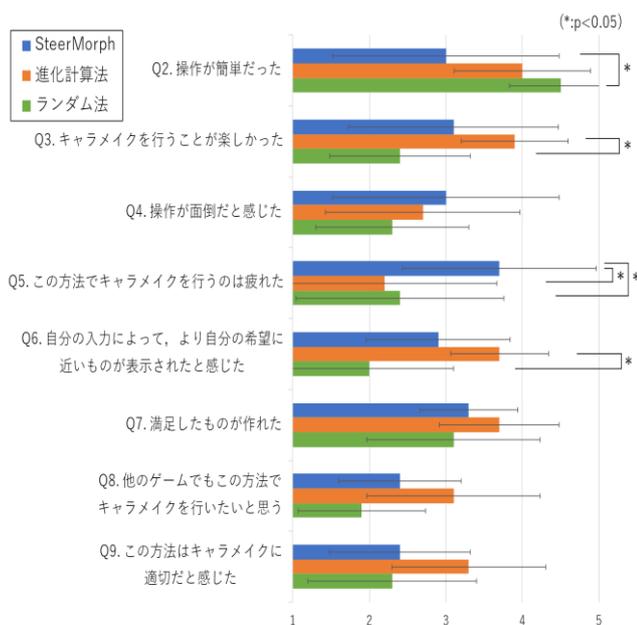


図7 実験後アンケートの結果

$$1 \text{ 回のタスク達成時間} = \frac{\text{タスク達成時間}}{(\text{リセット数}) + 1} [\text{分}]$$

この数式を適用すると、SteerMorph は 1.69、進化計算法は 2.15 という数値が算出される。導出した 1 回のタスク達成時間をランダム法のタスク達成時間と比較すると、1 回のタスク達成時間では同程度の時間でアバター作成が行えることが示唆される。

実験後アンケートの結果を図 7 に示す。グラフは 5 段階のリッカートスケールで得た値の各条件における平均値と標準偏差を示す。

実験後アンケートの得点について、一要因参加者内分散分析を行った (独立変数: 実験条件, 三水準 (SteerMorph, 進化計算法, ランダム法), 従属変数: アンケートの得点)。そのうち、群の効果に有意差が観察された項目について、Holm 法による多重比較を行った。操作の簡単さを問う質問 (Q2) では、SteerMorph の平均値がランダム法の平均値よりも有意に低いことが観察された ($p < .05$)。作業の楽しさを問う質問 (Q3) では、進化計算法の平均値がランダム法の平均値よりも有意に高いことが観察された ($p < .05$)。作業の疲労度を問う質問 (Q5) では、SteerMorph の平均値が進化計算法およびランダム法の平均値よりも有意に高いことが観察された ($p < .05$)。自分の入力によって、より自分の希望に近いものが表示されたかを問う質問 (Q6) では、進化計算法の平均値がランダム法の平均値よりも有意に高いことが観察された ($p < .05$)。

これらの結果により、SteerMorph は進化計算法、ランダム法に比べてキャラクターメイキングを行った後に疲労を感じたということがわかった。また、SteerMorph はランダム法に比べ、操作が簡単ではなかったことがわかった。進化計算法はランダム法と比較して作業が楽しく、自分の入力によってより自分の希望に近いものが表示されたことがわかった。なお、今回のアンケート評価ではどの手法がキャラクターメイキングに適切であるか明らかに出来なかった。

実験後の聞き取り調査では実環境を想定したときにどの手法を使いたいかに関して投票を行った。また各キャラクターメイキングの手法を使いたい順に順位付けを行った。その結果を表 3、表 4 に示す。実環境を想定したときにどの手法を使いたいかについての質問では SteerMorph は進化計算法と同票で、最も多い得票数であった。このことから SteerMorph はゲームで使用することを想定したときにも使ってみたいと思えた手法であることを確認した。手法ごとの順位と得票数では SteerMorph は票数にばらつきが見られた。このことから、参加者によって好みが見られたことや、改善可能性があることがわかった。

表3 実環境での使用を想定した質問と票数

もしもゲームに入っていたらどれを使いたいですか？ (複数回答可)	
手法	票数
SteerMorph	6
進化計算法	6
ランダム法	3

表4 手法ごとの順位と得票数

	各手法を順位付けすると、どの順番でキャラクターメイキングをしたいですか？		
	1位	2位	3位
SteerMorph	3	4	3
進化計算法	6	2	2
ランダム法	1	4	5

5. 議論

5.1 SteerMorph の操作感

実験後の聞き取り調査において参加者から『操作がシンプル』、『操作感は軽い』、『様々なものが出てくるのが良い』、『深く悩みすぎず、直感で入力を行える』といった評価が得られた。一方で画像の切り替えの早さから『即断即決がきつい』、『選択肢が多すぎて結果的に困った』、『スティックの傾きを用いた評価は慣れが必要である』などの意見も得られた。今回の実験において、参加者によっては短時間で切り替わる画像に対する評価が追いつかなかった様子が観察できた。こうした短時間で切り替わる画像を評価する行為の繰り返しが実験後アンケートにおけるキャラクターメイキング後の疲労や、操作が簡単とはいえないという結果に繋がったのではないかと考える。対策として参加者ごとに画像表示速度を調整することで見た画像と評価のタイミングがより合うようにすることができる。一方でアニメーションの切り替わりの気持ちよさや画像の評価の能率と画像の集束にかかる時間はトレードオフの関係にあるため、適切な表示速度に設定する必要がある。

実験後、参加者から『SteerMorph は漠然と探索するには向いているが、特定のものを目指して選ぶとしたときにどう選べばいいかわからない』という意見が得られた。これは SteerMorph の親の選択アルゴリズムについての理解が不十分であったことが原因として考えられる。本実装では親の選択アルゴリズムは各セグメント内で一番評価の高いものを親の候補とし、その後選ばれた親の候補を降順に入れ替える方式をとった。より明確に順位付けを行うためにはセグメント内で画像の評価は細かく分かれており、セグメント間は評価の差がある状態が望ましい。そのためスティック操作は傾きを細かく変えて、最大限右に傾ける操作は特に評価を行いたいときに限り使用する。これにより

順位付けが明確となり、特定のものを選びやすくすることができる。こうした操作面での理解を行うことで、より適応度を高めたキャラクターメイキングの実現が予期される。

5.2 SteerMorph の改善方法について

本実験では潜在ベクトルの変化に線形補間を用いたが、1 世代目は画像変化の振れ幅が特に大きく、評価をしづらいことが問題として挙げられる。対策として初期集団は 1 セグメント内の補間枚数を多くしておき、世代を重ねるごとに補間枚数を減らしていく。画像出力に用いる潜在ベクトルのベクトル空間の変動に伴う画像の大幅な変化を軽減することで、評価をやりやすくと予想される。あるいは画像の表示時間を評価が低いときは短くし、評価が高いときは長くするといった評価値によって画像の表示時間を変化させる方法も有効であると考えられる。

本実装では画像の評価を行うための入力インターフェースとしてゲームコントローラのアナログスティックを使用した。これはアナログスティックの傾きを入力することで、個々の画像に対して段階評価よりもさらに細かな解像度で画像の評価を行うことができると考えたためである。他の入力インターフェースとしてスロットルレバーやスライドボリューム、ロータリボリュームなどが挙げられる。入力インターフェースを変更することで、操作性や入力性能が向上する可能性がある。

一般的なゲームコントローラのアナログスティックは 2 軸の入力が行える。本実装では左右の 1 軸を気に入ったかどうかという指標で実装を行ったが、もう 1 軸を他の指標として用いることもできる。2 つの指標を 2 軸に適用することで、より効率の良いキャラクターメイキングが行える可能性がある。

本実験のパラメータにおいてモデルの見た目は 4 世代から 6 世代にかけて集束が行われた様子が確認できた。パラメータをより調節することで集束時間を短く、効率よくできる可能性がある。同様にランキング選択を行う際に固定値を用いたが、画像に対する評価値を用いてランキングの重みづけを行うことで、よりユーザの評価を反映できるのではないかと考えられる。また、対話型進化計算は初期集団によっては必ずしも評価の高い個体を作成できないという特性がある。そのため作りたいモデルのテーマがある場合、そのテーマに適した初期集団を作成することでより適応度の高い個体を組み合わせたキャラクターメイキングを行える。

5.3 SteerMorph の適用可能性

本研究では顔テクスチャや顔アイコンのための顔画像生成を行ったが、アクションゲームや RPG などで用いるための 3 次元アバターを生成するためには、使用するデータセットを変更する必要がある。具体的にはアバターの頂点データを学習させ、学習データを用いて 3 次元形状のモデルを生成する。このデータを用いて本手法を適用すること

で、3次元アバター作成ができると考えられる。また、本研究では顔画像データセットを用いてキャラクターメイキングを行ったが、他のデータセットを利用することもできる。例えば、車や動物、フォントなどに対して適用が見込まれる。

SteerMorphはモデルの探索や提示、およびその評価といった特性を生かすことで、初期集団のテーマ決定や完成されたモデルのアレンジといった適用可能性がある。例えば多様な種族が存在するゲームでは、ユーザはある種族を選択してゲームを始める方式がとられている。特定の種族に関心を寄せていないユーザにとっては実際に作成を行ったり、作成例を調べたりしなければどのようなものが作れるか分からないという問題がある。ここでSteerMorphによるキャラクターメイキングを行うことで、さまざまな顔の特徴が変化していく様子がアニメーションとして提示されるため、モデルの造形イメージをつかみながらアバター作成を行うことができる。モデルのアレンジは完成されたモデルに対して行う。本実験での実装はセグメント間において開始ベクトルは別になるようなベクトルの生成を行ったが、1セグメントおきに開始ベクトルを同一ベクトルにすることで、完成されたモデルとアレンジを行ったモデルを反復させることができる。この反復の中でより良いモデルを探索することができる可能性があると考えられる。

5.4 キャラクターメイキングはどうあるべきか

SteerMorphは前述のとおりアナログスティックを利用した操作の軽さや多種多様な画像が連続的に表示されていくことの良さが支持されていた。一方で画像の提示速度の調整やシステムの慣れが必要であるという意見が得られた。

進化計算法は『良い画像にめぐり合ったらそこに突き詰めていきやすい』、『提示される画像は一点に固まっていくため選択がしやすく、早い段階でイメージが固めやすい』などといった各画像に対してじっくりと吟味したうえで評価が行える点について好意的に受け止められていたものの、画像ごとに段階評価を行う操作の面倒さについての否定的な意見もあった。

ランダム法では『シンプルでよい』という操作の利便さについて支持されていたが、表示される画像に一貫性がないことや時間経過によって妥協してしまうといった意見があった。また本システムにおいて参加者は履歴が残らないことで次に良いものが出てくるか確証を得られないことに対する不安を感じていた。そのため履歴を残すことや、生成してきた中で一番良いものをストックして表示したいという意見が見られた。

実験後の聞き取り調査では静止画の表示枚数は6枚が適切であり、多くても10枚程度という意見が得られた。一方で収束によって画像の変化がなくなったことや時間経過といった理由からいずれの手法においても妥協を行ったことがあることがわかった。そのため作成後に手直しや微調整

を行いたいという意見も得られた。

聞き取り調査の中で、キャラクターメイキングの方針はおおむね2つのタイプに分けられることがうかがえた。明確に作りたいものが存在しているタイプと、作りながら考えていくタイプである。前者は理想に近づけやすい進化計算法がよく、後者は多様性のあるものの中から決定するランダム法が適していると考えられる。SteerMorphは画像の補間を行うことで多様性のある画像を提示しつつ、画像に評価を行うことである画像に近づけることもできるため、そのどちらにも対応することができる手法といえる。また、画像の更新に合わせて画像を評価するため、1軸のスティック操作を行うだけでアバター作成ができる。

本研究で作成した3手法はそれぞれ性質が異なるため、明確に作りたいものに寄せていきたいか、作りながら考えていくか、といったそのときの方針に合わせてキャラクターメイキングの手法を選ぶことで、より思い通りのデザインに近づけやすくすることができると思う。

6. おわりに

キャラクターメイキングの手法としてSteerMorphを提案した。実験はSteerMorph、進化計算法、ランダム法について顔画像におけるキャラクターメイキングを行い、アンケートによる評価実験を実施した。SteerMorphはタスク達成時間において作業効率の向上は果たせなかったが、1回のタスク達成時間では他の手法と同程度の時間でアバター作成が行えることが示唆された。アンケート評価の結果、SteerMorphは進化計算法、ランダム法と比較してタスク達成後に疲労を感じたことがわかった。またランダム法に比べ、操作が簡単ではなかったということがわかった。

今回のアンケート評価による結果では、どの手法がキャラクターメイキングに適切であるか明らかに出来なかったが、聞き取り調査においてSteerMorphはゲームで使いたいと感じた手法であることを確認した。また手法ごとに順位付けを行ったところ、票数にばらつきが見られ、参加者によって好みが分かれたことを確認した。こうした結果を踏まえて、SteerMorphの改善方法や適用可能性について検討を行った。

参考文献

- [1] Tero Karras, Samuli Laine, Timo Alia. A Style-Based Generator Architecture for Generative Adversarial Networks. CVPR, 2019
- [2] Hideyuki Takagi. Interactive Evolutionary Computation: Fusion of the Capacities of EC Optimization and Human Evaluation. Vol.89, No.9, pp. 1275-1296. IEEE, 2001.
- [3] Philip Bontrager, Wending Lin, Julian Togelius, Sebastian Risi. Deep Interactive Evolution. NIPS, 2017.
- [4] "MakeGirls.moe". <https://make.girls.moe/#> (参照 2020-1-29)
- [5] "WaifuLabs". <https://waifulabs.com> (参照 2020-1-29)
- [6] "Crypko". <https://crypko.ai/#/beta> (参照 2020-1-29)
- [7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio. Generative Adversarial

Nets. NIPS, 2014.

- [8] Hiroaki Nishino, Tsuneo Kagawa, and Kouichi Utsumiya. A Mobile Graphics System for Ubiquitous Environment. pp.83-90, NBiS, 2009.
- [9] 西野浩明, 青木研, 賀川経夫, 宇津宮孝一. 対話型進化計算に基づく3次元CG創作支援技術. 計測と制御, Vol.44, No.1, pp.16-22, 2005.