

オンライン会議カメラ映像のための個人差を考慮した顔面動作に基づく表情の可視化・増幅アバターシステムの検討

石出 宗己^{1,a)} 岩井 将行^{1,b)}

概要：現在の日本では総務省によるテレワークの推進で、ICT を活用し時間や場所に囚われない新しい働き方への取り組みが積極的に行われている。テレワークの導入でオンライン会議を行う機会が増えたが、参加者の一部がカメラをオフにしていると、お互いの顔が見えないまま会議を行うという現象が発生する。相手の顔が見えないと表情を見ることもできず、雰囲気をつまみづらい、遠慮が発生するといった影響が想定される。そもそも高解像度のカメラ自分の顔を取られ配信され続けるときに抵抗があるひとも多く、参加者のほとんどがカメラをオフにしている場合も少なくない。そこで本研究では、スマートフォンのカメラと 3D モデルを用いて表情データのみを可視化を行うシステムの開発を行った。カメラで取得した表情データを Facial Action Coding System に基づいて分析を行い、実際の表情として人型の 3D モデルに出力する。表情データには事前に個人に合わせて定義した倍率を設定し、明確に出力することが可能とする。さらに、正確に表情が伝わらない顔の作りや表情の出やすさにおける個人差をアバターを用いることで軽減させている。本システムが出力した表情データが表情として明確かつ有用であるかを検証し考察する。

1. はじめに

現在の日本では総務省によるテレワークの推進で、ICT を活用し時間や場所に囚われない新しい働き方への取り組みが積極的に行われている [1]。テレワークの導入において、オンライン会議を行うシステムを用いることは不可欠である。オンライン会議システムではカメラとマイクを用い、多くの人と擬似的に対面で会議することを実現している。しかし、参加者の一部がカメラを所有していないもしくはオフにしていると、お互いの顔が見えないまま会議を行うという現象が発生する。相手の顔が見えないと表情を見ることもできず、雰囲気をつまみづらい、遠慮が発生するといった影響が想定される。また、自分の顔が映るカメラを使用することに抵抗があるという理由から、積極的にカメラを使用しない人もいることがわかっている [7]。

本研究では、スマートフォンのカメラと 3D モデルを用いて表情データのみを可視化を行うシステムを提案する。本システムは、カメラで取得した表情データを Facial Action Coding System に基づいて分析を行い、実際の表情として人型の 3D モデルアバターに出力する。正確に表情が伝わらない顔の作りや表情の出やすさにおける個人差をアバターを用いることで、よりリアルな表情の出力を行っている [6]。表情データには事前に個人に合わせて定義した倍率を設定し、明確に出力することが可能となっている。

2. 関連研究

表情の認識分野では数多くの研究が行われている。表情の認識を行う指標として、P.Ekman らが提案した FACS (Facial Action Coding System) が用いられている。FACS は、顔面筋肉の動作を最小単位 ActionUnit (AU) に分類し、表情を客観的に記述する方法である [8]。山口らは、無表情の登録画像から、機械学習を用いて特定の表情を人工的に作り出し、追加の登録画像として用いることで認識精度を向上させるシステムを開発した [4]。機械学習で操作する顔のパーツとして、FACS の AU (Action Unit) にしたがって定義している。増田らは、AU にしたがってアニメーションを作成し、実際に撮影した表情の画像にモデルを変形させた場合とを比較、評価を行った [6]。坂本らは、通常表情の場合と無表情から表情記述を行う際に AU を用いた場合の比較を行った [5]。皮膚の皺以外では、十分に表情の表現が可能であることを示している。FACS では、人間らしい表情を作る際に有用であることがわかる。Dong らは、模擬面接時にカメラ映像をディープラーニングで分析し、面接評価を行うシステムを開発した [2]。モバイルアプリケーションで映像を記録し、専用のサーバーを介してリアルタイムなコーチングを行っている。システムでは 8

¹ 東京電機大学

^{a)} ishide@cps.im.dendai.ac.jp

^{b)} iwai@cps.im.dendai.ac.jp

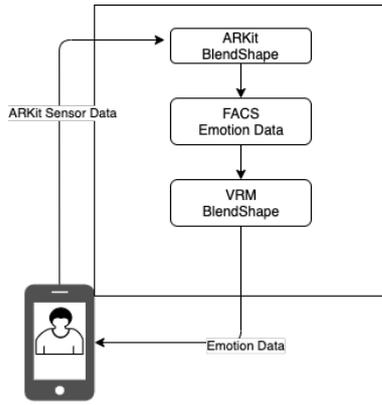


図 1 システム構成図

つの表情を認識していて、各値をグラフとして表示している。横山らは、FaceAPI を用いてカメラで取得した静止画を 5 秒毎に分析し、感情に応じた CG エフェクトを付加するアプリケーションを開発した [3]。表情を可視化することで自己表現や、第三者から見た自分を反省するために活用することができる。これらを両立することで、より有用性を求めることができると考える。本研究と類似した先行作品として、カメラで取得した顔に応じて画面内のキャラクターの表情を変化させる「FaceRig」が挙げられる。画像認識で検出した顔のパーツに応じて表情や顔の向きを変化させている。しかし、表情として変化させているのではなく、顔のパーツの動作をそのままキャラクターに割り当てているため、表情の出方に個人差が認められることが可能性として考えられる。

3. システム構成

本システムの構成図を図 1 に示す。本システムでは、スマートフォンのカメラを用いて 3 次元的に取得した特徴を、FACS に基づいた AU を参考にし紐付けを行って表情を分析する。分析時に表情値を算出し、3D モデルアバターに表情として適用する。表情値は事前に個人に合わせた倍率を計算しておき、表情適用時に増幅した値を利用する。スマートフォンは単体で動作し、個人に合わせた倍率は保存しておくことができる。3D モデルアバターは VRM 規格のものを使用し、表情の出力結果は VRMBlendShape の値の変化にしたがう。使用するスマートフォンは Apple の iPhone を使い、深度情報を取得可能な TrueDepthCamera で得た顔の 3 次元情報を ARKit を用いて取得している。実装には Unity を使用した。

4. 実装

この章では、本システムの動作についての具体的な実装内容についての説明を行う。

表情	Action Unit
喜び	4+5+25
悲しみ	6+12+26
驚き	1+4+15
怒り	1+2+5+27

表 1 FACS 表情と AU の対応

No.	ActionUnit
1	InnerBrowRaiser (眉の内側を上げる)
2	OuterBrowRaiser (眉の外側を上げる)
4	BrowLowerer (眉を下げる)
5	UpperLidRaiser (上まぶたを上げる)
6	CheekRaiser (頬を上げる)
12	LipCornerPuller (唇の端を上げる)
15	LipCornerDepressor (唇の端を下げる)
25	LipsPart (上下の唇を離す)
26	JawDrop (顎を下ろし、口を開ける)
27	MouthStretch (口を大きく開ける)

表 2 Action Unit の詳細

4.1 ARKit のパラメータを AU に変換する実装

本システムでは怒り、喜び、悲しみ、驚きの 4 表情の実装を行った。対応する FACS 表情と AU を表 1、表 2 に示す [9]。また、ARKit で取得可能なパラメータの一部をに

No.	BlendShapeLocation	No.	BlendShapeLocation
0	BrowDownLeft	17	EyeLookUpRight
1	BrowDownRight	24	JawOpen
2	BrowInnerUp	26	MouthClose
3	BrowOuterUpLeft	29	MouthFrownLeft
4	BrowOuterUpRight	30	MouthFrownRight
6	CheekSquintLeft	43	MouthSmileLeft
7	CheekSquintRight	44	MouthSmileRight
16	EyeLookUpLeft		

表 3 ARKit で検出できる顔面動作の一部

示す表 3[10]。FACS 表情として使用するために、ARKit で得られるパラメータを AU に変換する実装を行う。

表 2、表 3 を比較すると、ARKit では左右の区別をしているのに対し、今回使用する表情の AU では左右の区別をしていないため、ARKit で取得したパラメータの平均値を使用している。

表 2 の 25 は、表 3 は該当しないが、「顎を下へ動かさずに口を開く」と変換することができるため、表 3 の 24 と 26 が共に 0 である時が該当する。表 2 の 26 は、「顎を下ろし口を開く」ことから、表 3 の 24 が 1 であり 26 が 0 である時が該当する。表 2 の 27 は、「口を大きく開ける」ことから、表 3 の 26 が 0 である時が該当する。

以上で計算した値を、AU の値として表情の分析に使用している。

4.2 個人差を考慮した表情の出力の実装

FACS 表情 1 つに対応する AU はそれぞれ 3 つ以上が該当しているため、表情値を計算する際には AU それぞれを足し合わせた平均値を使用している。個人差を生む要因である無表情の値を、初期値として事前に記録する。また、表情の最大となる値も個人差を生む要因となるため、表情をそれぞれ一度ずつ作り値の記録を行う。本システムにおける表情値は 0 から 1 の値で表現している。最大時の表情値が 1 になるように倍率 x を計算するには、事前に記録した最大値を M とし、以下の式を用いる。

$$x = 1/M$$

最後に、個人差を考慮するための計算をまとめると、

$$E = E_c/M - E_0$$

となる。ここでは、 E は計算された現在の表情値、 E_c はデータから得た表情値、 M は表情の最大値、 E_0 は無表情時の表情値を表している。また、倍率をかけた表情値から初期値を引くことで、無表情時に生じていた表情間の誤差をなくす。すべての表情値を比較し、一番高いものを現在の表情として出力する。

4.3 アバターへの適用

表情として出力されたものを、VRM 規格で制作されたアバターへ適用する [11]。VRM は表情の管理を行うことができる他ヒューマノイドモデルの操作に特化したライブラリで、本システムでは表情出力するために採用した。得られた 0 から 1 までの現在の表情の表情値を該当する VRM の表情として出力している。

出力される表情例を図 2、図 3 に示す。



図 2 怒っている表情例

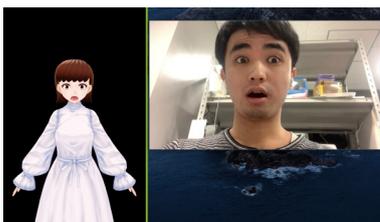


図 3 驚いている表情例

5. 実験結果考察

本システムの表情値の出力結果を検証するために、成人男性 3 名を対象として実験を行った。

5.1 実験の流れ

実験の流れは以下の通りである。

オンライン会議を開き、筆者がシステム利用者となりカメラとスマートフォンを用意する。システム利用者は 10 秒ごとに喜び、悲しみ、驚き、怒りの中からランダムに表情を変える。他の参加者は表情がどれに該当するかを推定し回答する。これをカメラで行った場合と、システムで行った場合で実施した。また、5 段階評価のアンケートも実施した。

5.2 実験結果

表情を推定する正答率はカメラを使用した場合と、本システムを使用した場合と同じで 9 割の正答率であった。アンケート結果を図 4、図 5 に示す。

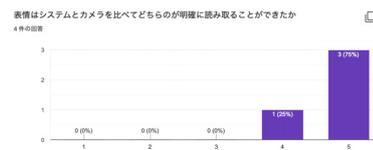


図 4 アンケートの結果 1

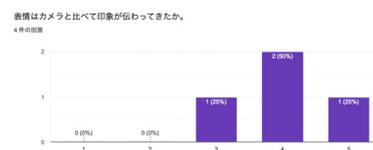


図 5 アンケートの結果 2

5.3 考察

正答率は変わっていないが、本システムを用いることで、表情の明確さを増幅させることができる可能性があることがわかった。また、「システムの方で表情がガタガタ変わってしまう点で惑わされることがあった。ここがもっと明瞭に変化が固定されていればもっと明確に読み取ることができ、良い判断材料になると思った。」という意見や、「悲しみと驚きはカメラごしの表情では判断が難しいことがある。このモデルでははっきりとした表情として見るので分かり易かった。」というフィードバックを得ることができたことから、安定性の改善を行うことで本システムは一定の有用性を得ることができると考えることができる。今後は、本システムの表情が変わりやすい不

安定な点の改善に加え、オンライン会議のより実践的な場面で使用できるよう機能の追加実装を行う。

参考文献

- [1] 総務省 HP:テレワークの推進, 入手先 [\(https://www.soumu.go.jp/main_sosiki/joho_tsusin/telework/\)](https://www.soumu.go.jp/main_sosiki/joho_tsusin/telework/)
- [2] Dong Hoon Shin, Kyungyong Chung and Roy C. Park: Detection of Emotion Using Multi-Block Deep Learning in a Self-Management Interview App, 入手先 [\(https://doi.org/10.3390/app9224830\)](https://doi.org/10.3390/app9224830) (2019).
- [3] 横山大希, 児玉幸子: Emotion FX:動画内の顔の感情をリアルタイムに強調する CG エフェクト自動生成アプリケーション, 入手先 [\(http://www.interaction-ipsj.org/proceedings/2020/data/pdf/1A10.pdf\)](http://www.interaction-ipsj.org/proceedings/2020/data/pdf/1A10.pdf) (2020.03.09).
- [4] 山口智美, 小田亮介, 上瀧剛, 内村圭一, 里中孝美, 牧岡毅: 表情変換を用いた本人認証に関する検討, 入手先 [\(https://www.jstage.jst.go.jp/article/itetr/35.52/0/35.52_1/_pdf/char/ja\)](https://www.jstage.jst.go.jp/article/itetr/35.52/0/35.52_1/_pdf/char/ja) (2012.12.12).
- [5] 坂本信樹, パクミンチョル, 苗村健, 原島博: FACSに基づく顔表情空間における表情記述と合成, 入手先 [\(https://www.jstage.jst.go.jp/article/iteac/1998/0/1998_281/_pdf/char/ja\)](https://www.jstage.jst.go.jp/article/iteac/1998/0/1998_281/_pdf/char/ja) (1998).
- [6] 増田隆, 高橋時市郎, 師井聡子: Facial Action Coding System に基づく表情アニメーションの評価, 入手先 [\(https://www.ieice.org/publications/conference-FIT-DVDs/FIT2007/pdf/I/L052.pdf\)](https://www.ieice.org/publications/conference-FIT-DVDs/FIT2007/pdf/I/L052.pdf) (2007).
- [7] コロナ禍オンライン会議増加で男性に変化3人に1人が美意識の高まり実感男性のほぼ半数が自分の顔を画面越しに見ることが「嫌だ」と回答, 入手先 [\(https://prtimes.jp/main/html/rd/p/000000058.000009540.html\)](https://prtimes.jp/main/html/rd/p/000000058.000009540.html) (2020.10.08).
- [8] 寺田員人, 花田晃治: 顔の表情とコンピュータグラフィックス, 入手先 [\(https://nds.dent.niigata-u.ac.jp/journal/301/t301_terada.pdf\)](https://nds.dent.niigata-u.ac.jp/journal/301/t301_terada.pdf) (2000).
- [9] 竹田智史, 岡田守: 表情豊かな手話アニメーションの生成, 入手先 [\(http://www.kochi-tech.ac.jp/library/ron/2001/info/1020299.pdf\)](http://www.kochi-tech.ac.jp/library/ron/2001/info/1020299.pdf) (2002.02.08).
- [10] Apple ARKit Documentation, 入手先 [\(https://developer.apple.com/documentation/arkit/arfaceanchor/blendshapelocation\)](https://developer.apple.com/documentation/arkit/arfaceanchor/blendshapelocation)
- [11] VRM - VR 向け 3D アバターファイルフォーマット -, 入手先 [\(https://vrm.dev\)](https://vrm.dev)