

2017 年度 卒 業 論 文

ポーカーにおける表情認識による
相手の手札推測に関する AI の研究

指導教員：渡辺 大地 准教授

メディア学部 ゲームサイエンス プロジェクト
学籍番号 M0114385
星 光彦

2018 年 3 月

2017年度 卒業論文概要

論文題目

ポーカーにおける表情認識による
相手の手札推測に関する AI の研究

メディア学部

学籍番号：M0114385

氏名

星光彦

指導
教員

渡辺 大地 准教授

キーワード

不完全情報ゲーム、ポーカー、表情認識、
AI、kinect

近年, AI に関する進化が著しい. 特にチェスや将棋, 囲碁などの AI は進化を続けており, チェスは 1997 年に AI である「Deep Blue」が世界チャンピオンに勝利した. その後もチェスに引き続き将棋の AI についても研究されており, 2010 年代からプロ棋士と対戦をするようになり, 2013 年にはプロ棋士に対して AI が圧勝している.

このように完全情報ゲームの AI が進化する裏で, ポーカーや麻雀などの完全情報ゲームとは対極を成す不完全情報ゲームの AI も進化を続けてきている. 特にポーカーに関しては 2017 年にカーネギーメロン大学の研究チームが開発した AI である「Libratus」がプロのプレイヤーに勝利している.

しかし, 不完全情報ゲームの AI はまだ課題を多く抱えており, 特にプロ以外の初心者や中級者のプレイヤーと楽しむプレイをする AI は少ない. そこで本研究では不完全情報ゲームで, 初心者や中級者がよく使用する「相手の表情から相手の手を推測する」テクニックに注目し, そのテクニックを利用した AI を研究することにした. 表情の認識には kinect を利用した. kinect とは, RGB カメラと赤外線カメラを搭載しているデバイスで, FaceTracking SDK を利用して顔の「目」「鼻」「口」などの位置を取得することができるものである. その kinect を利用し, プレイ中の相手の表情を認識し, その認識した表情から相手の手を推測した. その結果, 表情認識に関しては課題が残る形となってしまう, 相手の手を推測できたとは言えない結果となった.

目次

第 1 章	はじめに	1
1.1	研究目的と背景	1
1.2	論文構成	2
第 2 章	ポーカーと相手の手の推測について	3
2.1	ポーカーのルール	3
2.2	相手の手の推測	4
2.2.1	テル	4
2.2.2	本研究での推測することの定義	5
2.3	その他のポーカーの共通ルール	5
2.3.1	役の強さ	5
2.3.2	チップ	7
第 3 章	提案手法	9
3.1	提案手法とそのフロー	9
3.2	kinect for windows	10
3.3	AI の行動	16
3.3.1	手札交換の時の AI	17
3.3.2	チップ行動の時の AI	18
第 4 章	評価と分析	19
4.1	表情認識の検証と結果	19
4.2	相手の手を推測できるかの検証と結果	20
4.3	検証結果	22
4.4	考察	22
第 5 章	まとめ	23

謝辭 24

参考文献 25

目 次

2.1	ファイブカードドロワー	4
2.2	役の強さ	6
3.1	Kinect	10
3.2	手札が強い時	11
3.3	手札が弱い時	11
3.4	HD Face	13
3.5	3D Head Pose	14
3.6	真顔の時	15
3.7	困り顔の時	15
3.8	Y 座標の差	16

第 1 章

はじめに

1.1 研究目的と背景

近年, AI に関する進化が著しく, 特にチェスや将棋, 囲碁などの二人零和有限完全情報確定ゲームの AI は進化を続けており, チェスは 1997 年に AI である「DeepBlue」が世界チャンピオンに勝利した [1]. その後もチェスに引き続き将棋の AI についても研究されており, 2010 年代からプロ棋士と対戦をするようになり, 2013 年にはプロ棋士に対して AI が圧勝している [2]. また接待碁のような, 相手のレベルに合わせてプレイしていくといった, 初心者や中級者とプレイするための AI も存在する. 牛木ら [3] はこのようなゲームには同じ性質を持っていると述べており, その性質とは, ゲームには理論上必勝法と呼ばれるミニマックス法 (先手及び後手が最善を尽くせば, 先手必勝か後手必勝か引き分けになることが決まっていること) が存在することである. これは, AI にとって完全情報ゲームの方が扱いやすく, 不完全情報ゲームと比べて研究が進んだことを意味する. 不完全情報ゲームとは, ポーカーなどのカードゲームのように, ゲームのルールによってプレイヤー間の情報が全て共有されていないゲームの事である [4]. 完全情報ゲームとは違い, 相手の手を完全に読むことはできず, 開示されているわずかな情報からプレイしなくてはならないので, プロのプレイヤーであっても直感に頼ることがある. これらのゲームの AI では, ディープラーニング

(深層学習)を活用した「人間との対戦経験」からプレイ方法を学んでいく AI[5] や、モンテカルロ法を用いた AI[6] が代表的である。その理由としては、直感に頼る部分において、人間の打ってきた手を学習していくためである。しかし、これらの AI はプロのプレイヤーに勝利するための AI であり、初心者や中級者と楽しむようなプレイをする AI ではない。そこで、実際のポーカーゲームでも使われている「相手の表情から相手の手を予想する」テクニックに注目した。不完全情報ゲームにおいて、相手の手を予想するということはとても重要なテクニックであり、この予想が当たるか当たらないかが勝敗を決することも少なくない。また、相手の手を予想して行動することにより、直感のみで行動するよりも相手との駆け引き要素が強くなる。特にその中でも、相手の表情から相手の手を予想するテクニックは初心者や中級者でも利用されている。これらを踏まえ、相手の表情によって相手が有利か不利かを判断し、そこから自分の行動を決定する AI を作ることを目的として今回は表情認識の部分を実装した。表情認識には「Kinect for windows」を利用することにした。その理由として、kinect は顔の各点の座標を取得することができ、細かい顔の変化を追うには最適だからである。井川ら [7] は顔の各点の座標の変化を追い、表情認識を用いた顔文字入力システムを提案している。この kinect を利用し、「Facetracking」を用いた笑顔認識と、顔の各点の座標の動きから困り顔を認識する機能を実装した。そして、ポーカーゲームを表情認識を利用してプレイした場合の有効性について検証するために、実際に表情認識を利用してポーカーゲームをプレイした。その結果、表情認識に関しては課題が残る形となってしまう、ポーカーゲームで表情認識を利用した場合の有効性があるとは言えない結果になってしまった。

1.2 論文構成

本論文は全 5 章にて構成する。構成は 2 章にてポーカーと相手の手の推測について述べ、3 章では本研究での提案手法について述べる。そして 4 章で評価と分析について述べ、5 章でまとめについて述べる。

第 2 章

ポーカーと相手の手の推測について

本章ではポーカーのルールや、相手の手の推測について説明する.2.1 節では、本研究で採用するポーカーのルールについて説明する.2.2 節では、本研究での相手の手の推測について説明する.2.3 節では、その他のポーカーの共通ルールについて説明する.

2.1 ポーカーのルール

ポーカーのルールについてだが、同じくして不完全情報ゲームの AI を研究している Heinrich らの研究 [5] と同じ「テキサスホールデム」のルールではなく、「ファイブカードドロワー」のルール [8] を採用する. その理由として、テキサスホールデムよりファイブカードドロワーの方が不完全な情報が多いからである. ファイブカードドロワーのルールは、図 2.1 のようにお互いの手札は開示せず 5 枚であり、残りのカードを山札として裏向きにそのまま置く. その後、順番にプレイヤーは好きな数のカードを裏向きに場に置き、捨て札とする. その捨て札と同じ枚数山札からカードを引き、その 5 枚を自分の役として勝負するポーカーである. このように、ファイブカードドロワーのルールではお互いの手札 5 枚が全て開示されていないが、テキサスホールデムのルールではコミュニティカードと言って、お互いに使える 5 枚のカードが開示されている. 本研究では勝ち負けより相手の手を推測

することを重要視しているため、テキサスホールデムのルールの場合、相手の手を読む以外にも開示されている情報によって AI の思考が変化してしまう。それは本研究では適さないと考え、ファイブカードドロウを採用することにした。

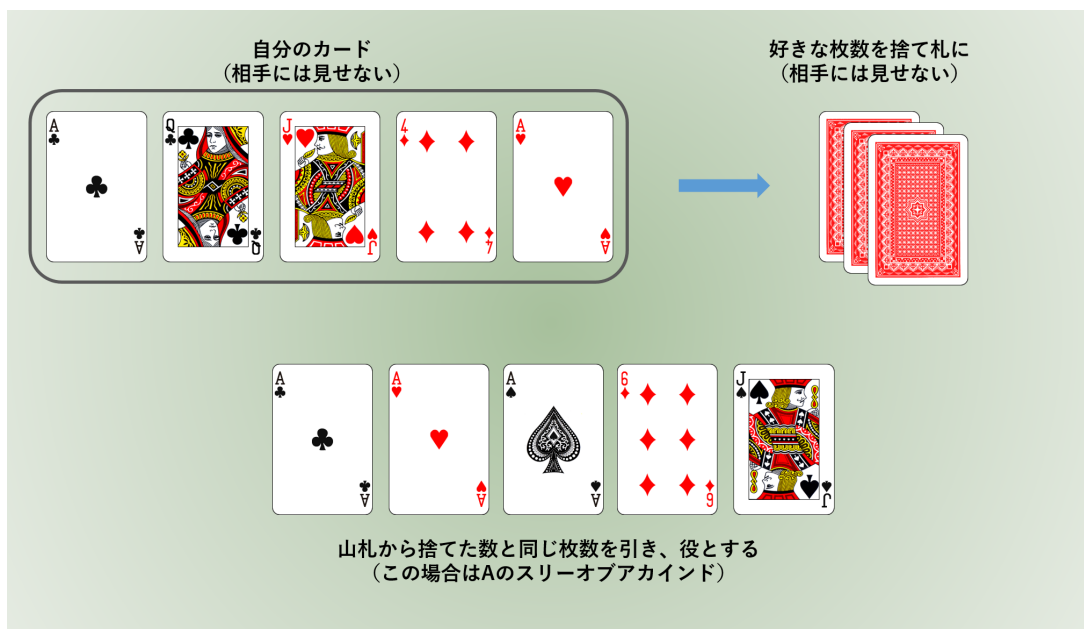


図 2.1 ファイブカードドロウ

2.2 相手の手の推測

この節では実際のポーカーゲームでも使われているテクニックである「テル」について 2.2.1 項で述べる。その後、2.2.2 項では本研究での推測することの定義について述べる。

2.2.1 テル

テルとは、「相手の表情や癖から相手の手を予想する」テクニックである [9]。表情は、手札によって感情が顔に出てしまうため、強かった場合には喜んだ表情に、弱かった場合には困惑したような表情になりやすい。癖とは、プレイ中に無意識に行ってしまう行動で、ベットサイズ（賭け金の大きさ）によるプレイに影響するものと、身体の動きによる癖の二種類がある。プロ同士のプレイでは

あまり使用されるテクニックではないが、プロではないプレイヤーは自分の感情が表情や行動によって表に出てしまうことも多く、ビギナー同士でのプレイではよく使用されるテクニックとなっている。

2.2.2 本研究での推測することの定義

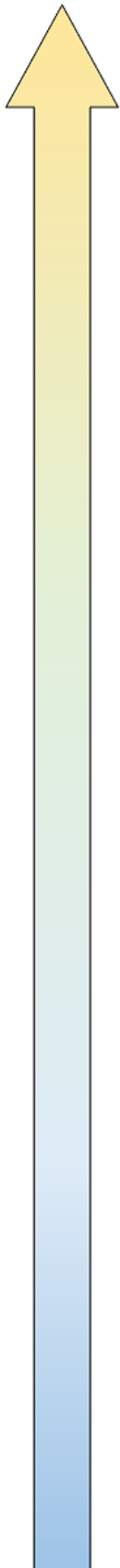
推測とは、与えられている情報を元に不完全な情報を予想することである。例えば七並びで、相手がハートの7より先を出さなかった場合、ハートの8を持っていないのではないかと予想することを推測という。本研究では、「ポーカーにおいて相手の手札が強いか弱いか予想すること」を相手の手を推測すると定義する。

2.3 その他のポーカーの共通ルール

この節ではその他のポーカーの共通ルールとして、AIに大きく関わる「役の強さ」と「チップ」について説明する。役の強さについては2.3.1項で述べる。チップについては2.3.2項で述べる。

2.3.1 役の強さ

ポーカーの役の強さは、図2.2が基本的な役の強さとなっており[10]、強い順に「ロイヤルフラッシュ（またはロイヤルストレートフラッシュ）」「ストレートフラッシュ」「フォーオブアカインド（またはフォーカード）」「フルハウス」「フラッシュ」「ストレート」「スリーオブアカインド（またはスリーカード）」「ツーペア」「ワンペア」「ノーペア」の順になっている。



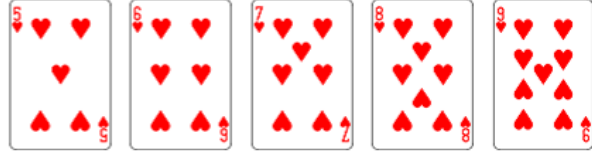
ロイヤルフラッシュ

同じスーツの
10, J, Q, K, A



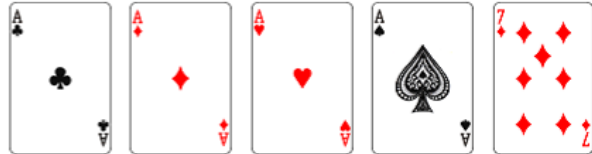
ストレートフラッシュ

同じスーツの
数字の並び



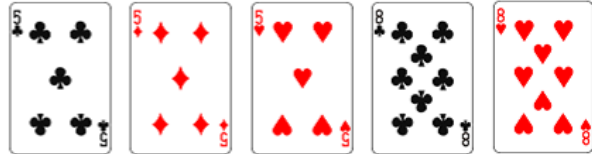
フォーオブアカインド

同じ数字の4枚



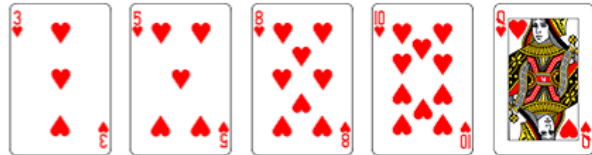
フルハウス

同じ数字の3枚と、
同じ数字の2枚



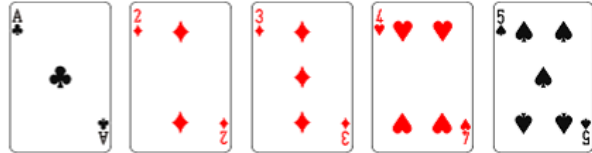
フラッシュ

全て同じスーツ



ストレート

数字の並び



スリーオブアカインド

同じ数字の3枚



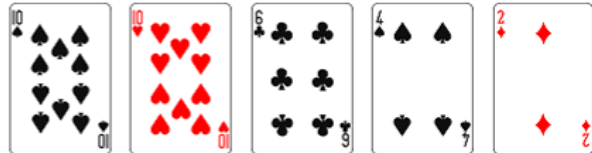
ツーペア

同じ数字の2枚が2組



ワンペア

同じ数字の2枚



ノーペア

数字もスーツもバラバラ

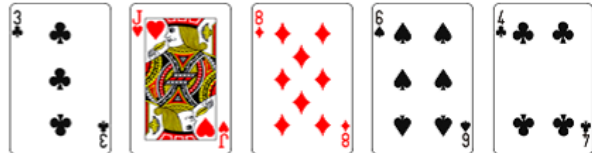


図 2.2 役の強さ

ロイヤルフラッシュが最も強く、ノーペアが一番弱い役となる。スーツとは、トランプのマークのことであり、強さの順は「スペード>ハート>ダイヤ>クローバー」となっている。さらに数字にも強さがあり、基本は数字が大きいほど強く、小さい方が弱い。しかし例外的に A が最も強い数字となっており、最も弱い数字は 2 である。対戦した時に相手と同じ役が出てしまった場合、その役を構成しているカードの中から 1 番強い数字を持っているプレイヤーの勝利となる。そこで本研究では役の中でどの数字が 1 番強いかわかりやすくするため、役の中で 1 番強い数字を先頭に持ってきて「K のワンペア」というように記述する。例えば数字が「2489Q」の場合、「Q のノーペア」と記述する。最も強い数字が同じだった場合はスーツの強さで比較する。

ファイブカードドロウのルールでの役の強さは、ツーペア以上が勝ちやすい役とされ、スリーオブアカインド以上が揃っていれば勝てる可能性が非常に高い。ファイブカードドロウのルールでは J 以上のワンペアで勝負することが多く、平均的な強さとなっている。10 以下のワンペアやノーペアは一般的に弱いとされる。また、この役の強さの順は強いものほど確率的に揃いづらいものとなっており、ロイヤルフラッシュやストレートフラッシュ、フォーオブアカインドが揃うことは稀である。そのため、実用的かつ強い役はフルハウスやフラッシュ、ストレートである。

2.3.2 チップ

カジノ等で行われるポーカーでは賭け金としてチップが使用され、最終的にこのチップを多く獲得したプレイヤーの勝利となる。チップの額は、ポーカーにおいて相手の強さを判断する重要な要素となっており、鬼沢ら [11] も相手のチップの額から自分のチップの額を決定するプレイングシステムを構築している。このチップのプレイ方法は大きく分けて二つあり、それぞれ「コール」「ベット」と呼ばれる。これらのプレイによって場にチップを出していき、勝った側が場に出ているチップを自分のものとする事ができる。

コールとは、相手と同じ枚数のチップを出すことで、自分の手札で相手に勝てるか不安な時や、多

くチップを失いたくない場合に使用する。例えば、自分の手札に揃っている役がワンペアの時、相手がノーペアか自分より弱いワンペアの場合でしか勝てる見込みがないため、コールを宣言することが多い。

ベットとは、相手より多くのチップを出すことであり、自分の札が強い時や勝ちたい時に使用する。自分が多くベットした時に相手はコールかベットを選ばなければならないため、必然と場に出すチップの枚数が多くなる。その状態で勝つことで多くのチップを手に入れることができるので、自分の役が強い時は積極的にベットをするのが重要である。

第 3 章

提案手法

本章では, 本研究での提案手法について述べる. 3.1 節では提案手法とそのフローについて述べる. 3.2 節では「kinect for windows」について述べる. 3.3 節では表情認識から手札を推測した後の AI の行動について述べる.

3.1 提案手法とそのフロー

本研究ではポーカーゲームにおける表情認識の影響を調査するため, ポーカーゲームを表情認識を行いながらプレイする. まずはプレイする 2 人のプレイヤーを, 表情認識を行いながらプレイをする「AI」側と, 普通にプレイをする「相手プレイヤー」側に分ける. 本研究では, AI 側は後にコンピュータ AI として実装することを前提としてるため, AI 側の行動はあらかじめ準備していた行動しかできないものとする. ポーカーゲーム開始から, AI からみて相手プレイヤーが最初に手札を見た時の表情を読み取り, その表情から相手の手を推測する. 次に, 推測した相手の手札を参考に AI 側の手札交換を行い, その後相手プレイヤーの手札交換も行う. 相手プレイヤーが手札交換をした後の表情も読み取り, 手札を推測する. この時に推測した相手の手札と AI 側の手札を比べ, そのデータをもとに AI 側のチップの行動を決定する. その後お互いの手札を開示し, 役が強かった方

の勝利とする. ここで, 本研究では勝敗よりも相手の手を推測できたことを重要視するため, 結果後の実際の相手の手札の強さと AI 側が推測した相手の手札の強さを比べ, その強さが近いレベルに達していることを目的とする. また, この目的を果たすため, 相手の表情を読み取るのに「Kinect for Windows」を用いた手法を提案する.

3.2 kinect for windows

Kinect は Microsoft 社から発売されているジェスチャーや音声認識によって操作ができるデバイスである [12]. RGB カメラと赤外線カメラが搭載されており, 画像と深度のデータを取得することが可能である. また, 画像と深度のデータより骨格のデータを取得することも可能である. そして, 「Kinect SDK 1.5」以降より「FaceTracking SDK」を利用して顔の「目」「鼻」「口」などの位置を取得することができる

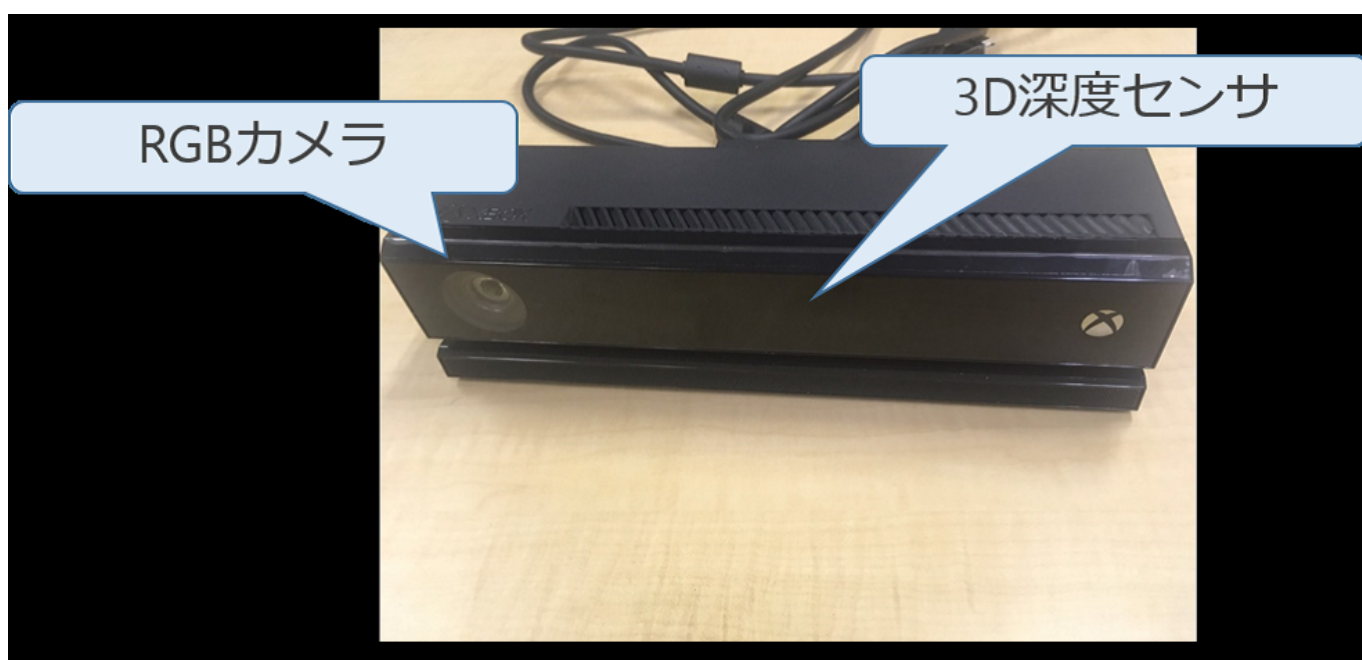


図 3.1 Kinect

本研究で kinect を利用して認識する表情は「笑顔」と「困り顔」の 2 つであり, 図 3.2 と 3.3 はその一例である. この 2 つの表情を認識する理由は, ポーカーゲームをプレイしているときの表情

の変化として, 手札が強い時に笑顔, 手札が弱い時に困り顔に変化するという大きく 2 つのパターンに分けることができるからである.



図 3.2 手札が強い時



図 3.3 手札が弱い時

表情認識を行うために「Kinect SDK v2」の「Face tracking」を利用する.Face trackingとは VisualC#で開発できるもので, 頭部の位置や傾きに加え, 色々な顔の情報を取得することができるものである. 静止画から読み取るのではなく, リアルタイムで顔を追跡できるのが特徴であり, 顔の情報を取得して, 呼吸を行っているかなども検知することができ, 吉武ら [13]はこの機能を利用した医療用監視システムを提案した. 取得できる情報は以下の 5 つである [14]

- 画像情報からの顔の検出
- 顔の部位各点の座標の取得
- 顔の向きの取得
- 顔の 3D モデルの取得
- 色々な顔の情報 (FaceProperties)

笑顔認識のために、「FaceProperties」を利用する.FaceProperties とは、以下の 8 つの顔の情報を Yes,Maybe,No と、測定不能状態の Unknown の合計 4 段階の値で取得できるものである [15].

- Happy(笑顔かどうか)
- Engaged(こちらを見ているか)
- WearingGlasses(眼鏡をかけているか)
- LeftEyeClosed(左目を閉じているか)
- RightEyeClosed(右目を閉じているか)
- MouthOpen(口を開いているか)
- MouthMoved(口が動いているか)
- LookingAway(目をそらしているか)

この値を利用し、「Happy」の値が Yes の場合に、笑顔と認識する.FaceProperties で困り顔の判断は不可能だが、困り顔の時に口が開いていることは無いため、MouthOpen の値が NO であることが困り顔の 1 つの条件とできる。さらに、今回の研究では Kinect SDK v2 の「HD Face API」を利用する.HD Face API とは高度な顔追跡ライブラリであり、人間の顔を検出するだけでなく 3D 空間内の 1,000 以上の顔の点にリアルタイムでアクセスできるものである [16]. 図 3.4 は実際に顔の各点を青い点で描画したものである。これらの点は全て固定のインデックス番号が振られており、例えば左眉毛のインデックス番号は 804,838,858,955,956,957 の 6 つで構成されている [17].

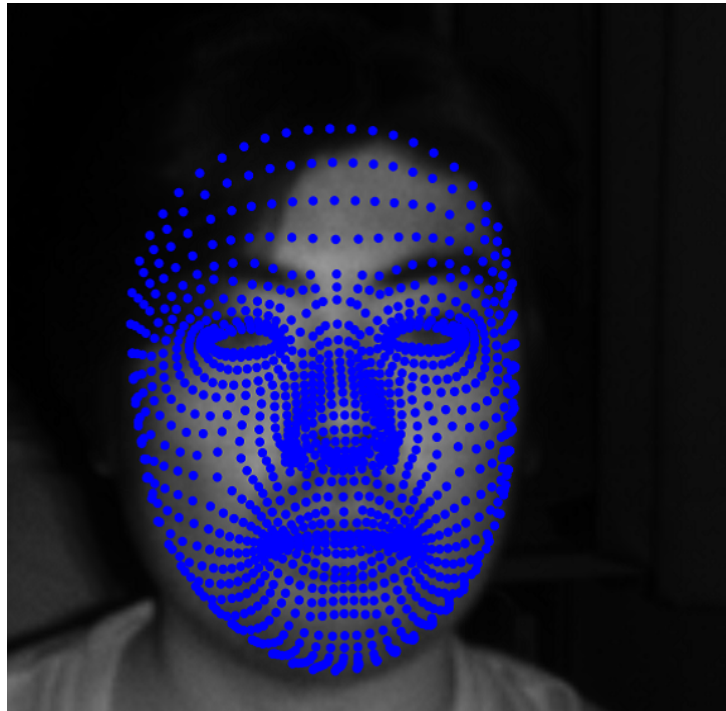


図 3.4 HD Face

だが,HD Face API は点の座標を 2 次元座標で取得しているため,顔の向きが真正面を向いていない場合にズレが生じてしまう.そこで,「3D Head Pose」を利用する.3D Head Pose とは,頭の角度をそれぞれ *pitch* 角,*roll* 角,*yaw* 角で表したものである [18]. これらの角度は度数法で表され,-180 度から +180 度の範囲の値で表される.*pitch* 角は上下の向きを表しており,-90 度で床に向かって見下ろしている状態を表し,+90 度の値で天井を向いている状態を表す.また,10 度未満の時に顔の追跡が最も効果的に行われる.*roll* 角は顔の傾きを表しており,-90 度で右肩と水平になる状態を表し,+90 度で左肩と水平になる状態を表す.*roll* 角は 45 度未満の時に顔の追跡が最も効果的に行われる.*yaw* 角は左右の向きを表しており,-90 度で右肩に向かって向いている状態を表し,+90 度で左肩に向かって向いている状態を表しており,30 度未満の時に顔の追跡が最も効果的に行われる.図 3.5 はこれらの顔の角度の模式図である.

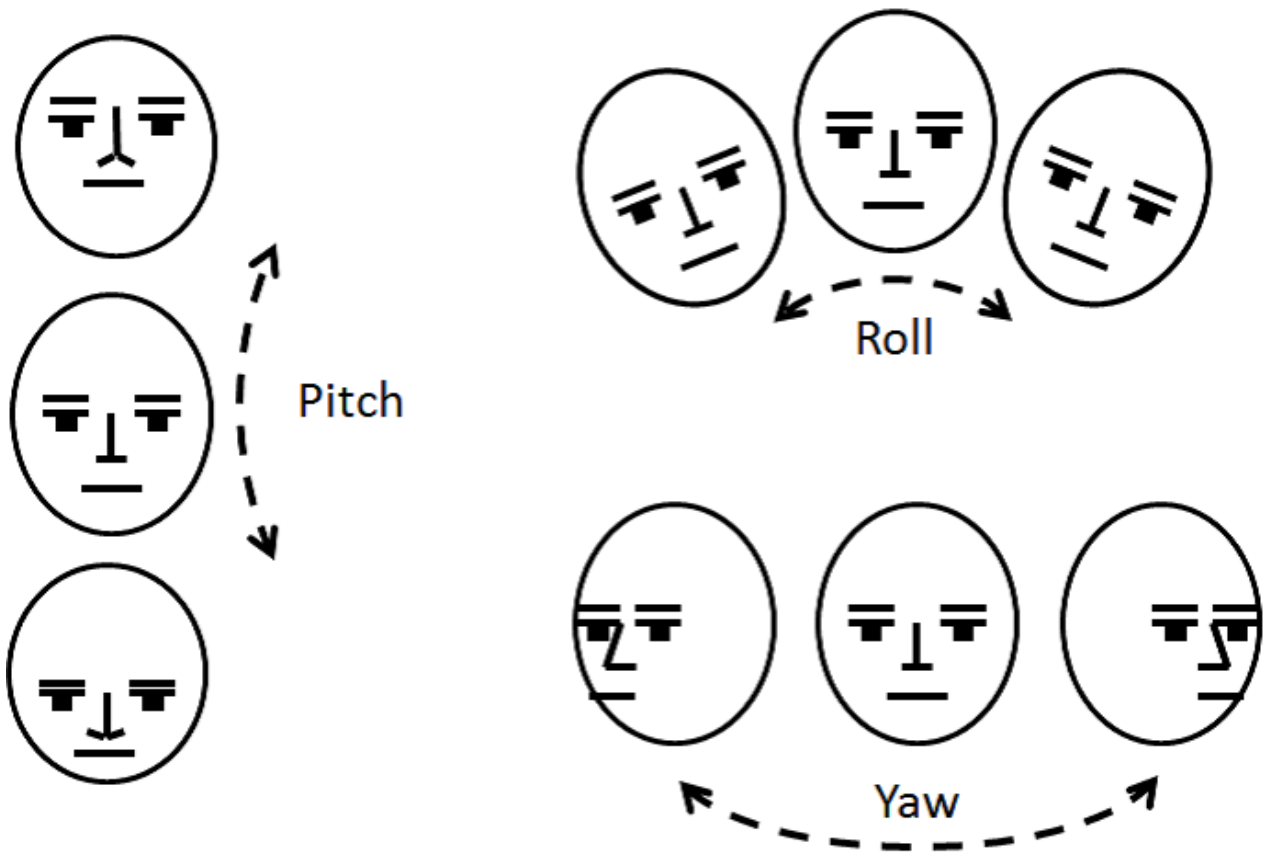


図 3.5 3D Head Pose

これらを踏まえ、「 $|pitch| < 10, |roll| < 45, |yaw| < 30$ 」の値を取っている間のみ HD Face API の顔の追跡を行う。

困り顔の認識は、眉毛の点の座標変化で行う。普段の眉毛はへの字だが、困り顔になったときは斜めに一直線の形になる傾向が高い。そこで両眉毛の両端の座標と、真ん中の座標を取得し、これらを二次元座標として配置した時の Y 座標の変化を追う。この時、左眉毛を真正面からみて一番左のインデックス番号 804 の点を点 **L**、真ん中の 955 番を点 **M**、一番右の 957 番を **R** とする。式 (3.1) は、**M** の Y 座標である M_y と **L** の Y 座標である L_y の差を p として表している。同様に、式 (3.2) は R_y と M_y の差を q として表している。

$$p = M_y - L_y \tag{3.1}$$

$$q = R_y - M_y \quad (3.2)$$

困り顔と判断する場合は、フレームごとの p の値と q の値を比較し、値が一定以下になった場合に判断を行う。フレームとは、動画を構成する 1 枚 1 枚の静止画 (コマ) のことである [19]。動画を表示する際には、極めて短い一定の時間間隔で次々に静止画像を上書きすることで人間の目に動いているように見せているが、この 1 枚ずつの静止画像をフレームという。fps とは、動画の滑らかさの指標として、1 秒間に書き換えるフレームの数を表す単位のことであり、今回は 30fps で計算した。

また、左眉毛と同様に右眉毛のインデックス番号についても調べ、それらの重要なインデックス番号の Y 座標同士も比較する。両眉毛の重要な点を別の色で描画し、真顔をしているのが図 3.6 であり、実際に困り顔をしたのが図 3.7 である。また、その時の左眉毛の各点の Y 座標の変化を、 $\text{point_distance} = (p, q)$ と表したものが図 3.8 である。

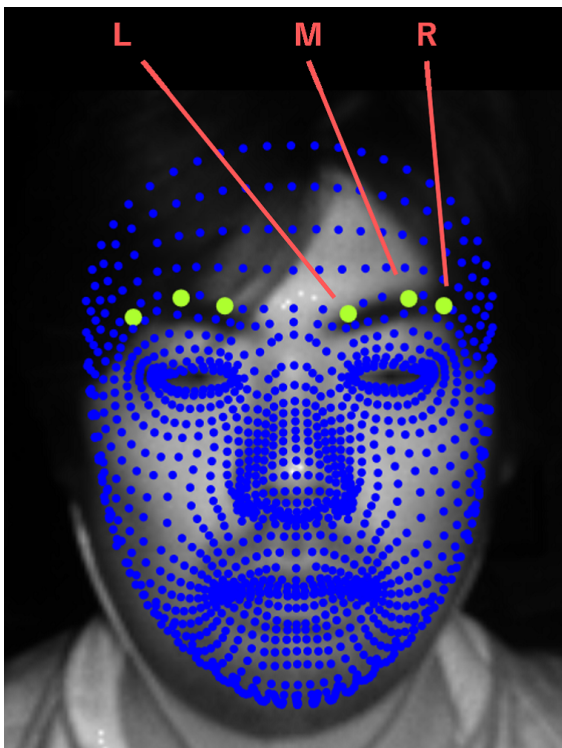


図 3.6 真顔の時

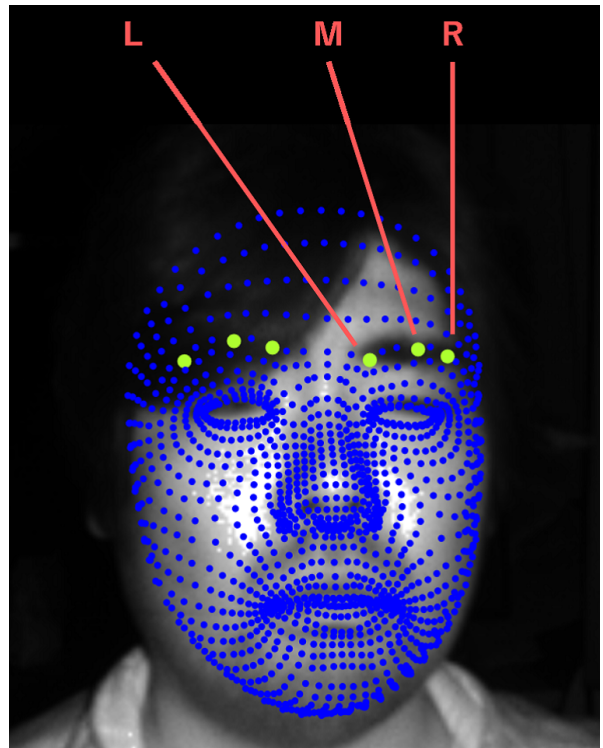


図 3.7 困り顔の時



図 3.8 Y 座標の差

真顔の時と困り顔の時の p の平均はおよそ 1.40 と 0.83 であり, その差は約 0.57 である. 同じように q の平均はそれぞれおよそ 1.74 と 2.27 であり, その差はおよそ 0.53 である. そこで, p の値が前フレームより 0.5 以上の差があり, かつ q の値も前フレームより 0.5 以上の差がある場合困り顔と判断する. 式 (3.3) は, 変数 t を現在のフレームとし, 前フレームの p の値を p_{t-1} , 前フレームの q の値を q_{t-1} とした場合に困り顔と判断する条件式である.

$$0.5 \leq |p_{t-1} - p_t| \cap 0.5 \leq |q_{t-1} - q_t| \quad (3.3)$$

3.3 AI の行動

本研究では AI 側の行動をプレイヤーが行うため, そのプレイヤーの行動を AI と同じ仕組みで決める必要がある. そこで, 表情認識をした後に AI が行動するパターンをいくつか設定し, その通りに AI 側のプレイヤーはプレイをする. この時, AI 側の行動を表情認識した際に相手の表情を笑顔と認識した場合に行う行動, 相手の表情を困り顔と認識した場合に行う行動, 相手の表情に変化がなく, 真顔と認識した場合に行う行動の 3 つのパターンに分けることができる. また, 表情認識は, 最初に手札が配られて手札交換する間に 1 回と, 交換し終わってからチップの行動をする間に

1回の計2回行う。また、表情認識の後に行うAI側の行動は、最初の1回と次の2回目では別の行動を決定する。本節では、手札交換に関するAI側の行動については3.3.1項で述べ、チップの行動を決定するAI側の行動についてを3.3.2項で述べる。

3.3.1 手札交換の時のAI

まず一番最初に手札交換を始める前、相手が配られた5枚のカードを見た時に表情認識を行う。その時に得た結果からAI側の手札交換が変化する。相手の表情に変化がなく、真顔と認識した場合には、今手札にある役を崩さないように残りのカードを捨て、1ランク上の役を目指すよう行動する。また、手札に役がなかった場合、その中で最も強いカード1枚だけ残してそれ以外を捨てる。これは、もし役が揃わずノーペアだった場合に最も強いカードを残しておくことで、相手もノーペアだった時に少しでも勝率を上げるためである。

次に、表情認識した際に相手の表情が笑顔だった場合、AI側はストレート以上の役を揃えることを目的として手札交換を行う。しかし、ロイヤルフラッシュやストレートフラッシュは揃う確率が低いため、この2つは目的としない。まず、連続する数字が3つ以上ある場合はストレートを狙う。この連続する数字が3つ以上ある場合とは、「AA234」のように「A234」の部分が連続している場合を表す。この場合は連続している数字以外のカードを全て捨て、ストレートのみを目指すよう行動する。「AA234」の場合は、「A」2枚のワンペアを残さずに「A」1枚を捨てる。また、数字が連続していない場合でも、間に数字が入ることでストレートが揃う場合、交換枚数が2枚以下で済む場合のみ手札交換を行う。例えば「357JQ」のように、「357」の間に「4」「6」といった数字が入ることでストレートが揃う場合、「JQ」を捨てる行動を取る。次に、同じスーツが3つ以上ある場合はフラッシュを狙う。同じスーツが3つ以上ある場合とは、数字が「A479J」のようにバラバラでも、スーツが「ハート、ハート、ハート、スペード、クローバー」のように同じスーツが3つ以上かぶっていた場合のことである。この場合はかぶっているスーツ以外のカードを捨て、フラッ

シュを狙う。今回のように「A479J」,「ハート,ハート,ハート,スペード,クローバー」となっている場合は,スペードとクローバーの「9J」を捨てる行動を取る。また,最初からツーペアかスリーカードが揃っている場合は,ストレートやフラッシュを狙わずにフルハウスを狙うことを目的とする。この場合は,役を残しながら,役を構成していない余りのカードを捨てる。例えば「AA225」の場合はツーペアを構成している「AA22」は残し,「5」1枚を捨て,フルハウスを狙うように行動する。「AAA58」のように最初からスリーカードが揃っている場合では「58」を捨てる行動を取る。「2579Q」,「ハート,スペード,スペード,クローバー,ダイヤ」のように,2枚以下の交換でストレート以上の強力な役を作ることが困難な場合は,相手の表情を真顔と認識した時と同じく,ワンランク上の役が揃うように手札を交換する。

最後に,相手の表情を困り顔と認識した場合,この場合も相手の表情を真顔と認識した時と同じく,ワンランク上の役が揃うように手札を交換する。

3.3.2 チップ行動の時の AI

チップ行動を決定する時の AI の行動は,相手の表情が笑顔の場合はコールを基本的に行う。次に,相手の表情が困り顔の時はベットを行う。相手の表情を真顔と認識した際に,スリーオブアカインド以上の役を持っている場合はベットを行い,ツーペア以下の場合はコールを行う。また,特殊なパターンとして,AI 側の手札がストレートを超える場合は勝てる可能性が高いため相手の表情に関わらずベットを行う。

第 4 章

評価と分析

本章では本研究の提案手法を評価するために行った検証について述べる。本研究では、まず表情認識の正確性について検証した。その後、ポーカーゲームにおいて、表情認識をすることで相手の手を推測できるか検証した。4.1 節では表情認識の検証と結果について、4.2 節では相手の手を推測できるかの検証と結果について述べる。

4.1 表情認識の検証と結果

表情認識の正確性について検証するために、実際に被験者に笑顔と困り顔をしてもらった。その顔を 3.2 節で示した提案手法通りに認識を行い、表情認識の判定が正確に行われているかデータを取った。被験者には最初に笑顔をしてもらい、その時の表情を「笑顔」か「困り顔」かを認識した。また、そのどちらにも属さず、認識できなかった場合に「真顔」とし、計 3 つの表情に結果を分けた。被験者に笑顔をしてもらった後に困り顔をしてもらい、同じく 3 つの表情に結果を分けた。今回の検証では 20 歳から 22 歳の男性 10 人に対して検証を行い、以下のデータを表にまとめる。

- 被験者 NO
- 笑顔をしてもらった時に認識した表情

- 困り顔をしてもらった時に認識した表情

表 4.1 は表情認識の結果である.

表 4.1 表情認識の検証

被験者 NO	笑顔時	困り顔時
NO.1	真顔	困り顔
NO.2	笑顔	困り顔
NO.3	笑顔	笑顔
NO.4	笑顔	困り顔
NO.5	笑顔	笑顔
NO.6	笑顔	困り顔
NO.7	笑顔	困り顔
NO.8	笑顔	困り顔
NO.9	笑顔	真顔
NO.10	笑顔	困り顔

表 4.1 の笑顔時に笑顔と判定した結果は 10 回中 9 回で,90% の確率で認識することができた. また, 困り顔の判定は 10 回中 7 回で,70% の確率で認識することができた. また, 困り顔時に笑顔と認識してしまった時が 2 回あった. 全体での認識の正答率は 80% ほどであった.

4.2 相手の手を推測できるかの検証と結果

検証を行うにあたってファイブカードドローレールのポーカーゲーム中に表情認識を行った. 表情認識の結果は 4.1 節と同じく「笑顔」「困り顔」「真顔」の 3 つに分けた.kinect を使って表情認識をしながらプレイを行う側を AI 側,AI 側からみて相手プレイヤーを被験者とする. また,AI 側のプレイは 3.3 節で示した AI の仕組み通りにプレイを行う. トランプはジョーカーを除いた 52 枚で行い, チップは 5 枚とする. 今回の検証ではチップの金額は関係ないため,1 枚の金額は不定とし, 枚数のみで計算を行う. まず,kinect で相手プレイヤーの顔を認識する. 認識できたら場代としてお互いのプレイヤーはチップを 1 枚出す. その後カードを配り, 相手プレイヤーにカードを手にして手札を見てもらう. この時, 手札を見ている間の表情を「表情 A」とし, この表情 A を表情認

識する. その後, 表情 A の表情認識を元に AI 側の手札交換を行う. 次に相手が手札交換を行う. この時, 相手が手札交換を終わった後の表情を「表情 B」とし, この表情 B も表情認識を行う. 次に相手側のチップを賭けるターン, ベットラウンドを行う. 相手は「コール」か「ベット」を選び, チップを賭ける. 相手のベットラウンドが終わったら表情 B の表情認識を元に AI 側のベットラウンドを行う. このベットラウンド終了後, お互いの手札をオープンし, 勝敗を決める.

今回の検証では相手の手札を AI 側の推測と一致していたかを評価する. AI 側の推測は, 相手の表情を認識した時に 4.2 の表の通りに推測する.

表 4.2 表情から推測する強さ

表情 A \ 表情 B	笑顔	困り顔	真顔
笑顔	強い	普通	普通
困り顔	やや強い	弱い	弱い
真顔	やや強い	弱い	判断不能

この 9 つのパターンで判断し, この判断が的確なのかを, 相手の手札を試合終了後に確認し, AI 側の推測と一致しているか評価する. また, 今回の検証では 20 歳から 22 歳の男性 4 人と女性 1 人に対して 2 試合ずつプレイし, 以下の結果を表にまとめる.

- 表情 A の表情
- 表情 B の表情
- AI が推測した相手の役の強さ
- 実際の相手の役
- AI 側の役
- 勝敗

4.3 検証結果

3.3.2 節の検証結果を表 4.3 にまとめた.

表 4.3 相手の手を推測できるかの検証結果

試合 NO	表情 A	表情 B	推測した強さ	相手の役	AI 側の役	勝敗
NO.1	笑顔	真顔	普通	J のノーペア	K のノーペア	AI の勝ち
NO.2	真顔	笑顔	やや強い	9 のワンペア	A のワンペア	AI の勝ち
NO.3	真顔	笑顔	やや強い	2 と K のツーペア	K のノーペア	AI の負け
NO.4	困り顔	真顔	弱い	7 のワンペア	K のワンペア	AI の勝ち
NO.5	真顔	真顔	判断不能	10 のワンペア	A のノーペア	AI の負け
NO.6	真顔	笑顔	やや強い	7 のノーペア	A のノーペア	AI の勝ち
NO.7	真顔	笑顔	やや強い	A のワンペア	A のノーペア	AI の負け
NO.8	真顔	真顔	判断不能	K のノーペア	A のノーペア	AI の勝ち
NO.9	笑顔	真顔	普通	Q のワンペア	A のスリーカード	AI の勝ち
NO.10	真顔	笑顔	やや強い	4 と 10 のツーペア	Q のノーペア	AI の負け

表 4.3 の AI が推測した相手の役の強さと実際の相手の役が一致した試合 NO は「NO.3」「NO.7」「NO.10」であり,30% 程の確率で推測できた. その 3 回のうち,AI 側が勝てた試合は 1 回もない. 表情を認識する回数が 20 回中真顔と認識した回数が計 12 回あり, 表情 A,B の双方とも真顔と認識して強さが判断不能になった試合が 2 つあった. また, 唯一困り顔と認識した「NO.4」は推測した強さと相手の役の強さが一致しなかった.

4.4 考察

相手の手を推測できた確率は 30% 程であるが, この結果に有意水準を 5% で二項検定を行うと, 推測できたとは言えない結果になった. この原因は, 今回の研究では困り顔の認識は kinect にもともと存在する機能を使用せずに行ったため, 笑顔認識との判断率に 20% 程の差が出てしまったことが挙げられる. また, 真顔を多く認識してしまった事も相手の手を推測できなかった原因の 1 つで, 表情認識の精度だけでなく, ポーカーゲームをやるうえで相手が真顔でプレイするのが要因だと考察できる.

第 5 章

まとめ

本研究では, 不完全情報ゲーム AI での駆け引き要素を強めるため, 表情認識をして相手の手を推測する研究を行った. 本研究の提案手法では表情認識を行って相手の手を推測することが可能であるかどうかを検証した. 今回の検証結果から, 表情認識で相手の手を推測することができず, 表情認識の精度について課題残った.

本研究で実装した表情認識の精度は高いとはいえず, 困り顔の認識の精度が低かった. また, ポーカーゲーム中では相手プレイヤーが意識して真顔をすることが多く存在したため, 表情認識をすることができなかった場面も多数存在した.

今後の展望としては, 表情認識にディープラーニングを実装する. 各試合ごとの顔の各点の座標を記録し, その各試合での相手の手札の強さも記録する表情認識にする. その後, 相手の顔の各点の座標が, 記録した各点の座標のデータと一致した場合, その一致した場合の試合の時に記録した相手の手札の強さを参考にし, 推測するようにする. また, 表情だけではなく, 癖も参考にし, 相手がずっと真顔をしていた場合でも推測できる状態を作る.

なお, 本論文は, 第 17 回デジタルコンテンツクリエイションにおいて”ポーカーにおける表情から相手の手を見抜く AI” [20] として発表した内容を含む.

謝辞

本研究を行うにあたって，渡辺大地准教授，三上浩司教授，阿部雅樹実験助手に心から感謝いたします。また、研究室の方々，実験に協力をしてくださった方々にも同じく感謝いたします。皆様本当にありがとうございました

参考文献

- [1] Murray Campbell and A. Joseph Hoane Jr. Deep blue. *Artificial intelligence*, Vol. 134, pp. 57–83, 2002-01-01.
- [2] 松原仁. ゲーム情報学：コンピューター将棋を超えて. 情報管理, Vol. 59, pp. 89–95, 2016-05-01.
- [3] 牛木美歩, 中西敏彰. ゲーム開発における思考ルーチンについての一考. 日本大学工学部学術講演会予稿集, Vol. 53, pp. ROMBUNNO.J-46, 2009.
- [4] 西野哲郎. 不完全情報ゲーム. 情報処理学会, Vol. 53, pp. 112–117, 2012.
- [5] Johannes Heinrich and David Silver. Deep reinforcement learning from self-play in imperfect-information games. *University College London, UK*, Vol. 10, pp. 1–10, 2016-06-28.
- [6] 松岡確, 堀内研, 中山泰一. 不完全情報ゲーム uno のモンテカルロ法による解法. 情報処理学会, Vol. 76, pp. 605–606, 2014.
- [7] 井川大輔, 笹岡久行. Kinect を用いた表情による入力インターフェースの提案. 情報処理学会, Vol. 76, pp. 109–110, 2014.
- [8] ファイブ カード ドロー - ポーカー用語集 — pokerstrategy.com. <https://ja.pokerstrategy.com/>

pokerstrategy.com/glossary/_519/. 参照:2017.7.18.

- [9] テル (癖) — ポーカー道. <http://www.pokerdou.com/texasholdem-beginning/tell/>. 参照:2017.5.25.
- [10] ポーカーの役について「役一覧表」: エベレストポーカーフォックスの攻略記録. <http://pokerfox.seesaa.net/article/268020802.html>. 参照:2017.11.3.
- [11] 鬼沢武久, 嵐見覚, 高橋千晴. 不完全情報ゲームプレイングシステムの構築. 日本知能情報ファジィ学会誌, Vol. 15, pp. 127–141, 2003.
- [12] Kinect - windows アプリの開発. <https://developer.microsoft.com/ja-jp/windows/kinect>. 参照:2017.7.18.
- [13] 吉武伸泰, 恒田晃完, 灘口尚大, 田中康一郎. Kinect センサを用いた医療用患者監視システムにおける体勢検知機能の実装. 情報処理学会, pp. 1–8, 2013.
- [14] Kinect for windows v2: face tracking 入門 - build insider. <https://www.buildinsider.net/small/kinect2fun/01>. 参照:2017.12.03.
- [15] Kinect for windows v2 で face tracking をします (その 2-2). <http://lab.aquaring.co.jp/archives/613>. 参照:2017.11.3.
- [16] High definition face tracking. <https://msdn.microsoft.com/ja-jp/library/dn785525.aspx?f=255&MSPPErrror=-2147217396>. 参照:2018.1.10.
- [17] How to use kinect hd face — vangos pterneas. <https://pterneas.com/2015/06/06/kinect-hd-face/>. 参照:2017.12.5.
- [18] Face tracking. <https://msdn.microsoft.com/en-us/library/jj130970.aspx?f=255&MSPPErrror=-2147217396>. 参照:2018.1.10.
- [19] フレームとは - it 用語辞典. <https://goo.gl/dWYixN>. 参照:2018.1.10.
- [20] 星光彦, 阿部雅樹, 渡辺大地, 三上浩司. ポーカーにおける表情から相手の手を見抜く AI. 情

報処理学会第 17 回研究報告デジタルコンテンツクリエーション (DCC) , Vol. 2017-DCC-17(12), pp. 1-2, 2017.