

筑波大学大学院博士課程

システム情報工学研究科修士論文

仮想空間への没入とハンドトラッキングを用いた遠隔会議システム

根岸 匠

修士（工学）

(コンピュータサイエンス専攻)

指導教員 田中 二郎

2016年3月

## 概要

遠隔会議は高速なインターネット回線やビデオチャットをするのに十分な性能を持ち、かつ低コストであるコンピュータが一般に普及したことを要因として広く用いられている。インターネットを介して遠隔で相手と意思疎通や会議ができるほか、移動に伴う経費や時間の削減に有効であるなどの利点があることから実際の会議の代替として用いる事例が増えてきた。

現在はビデオチャットを用いた遠隔会議システムが一般に普及しているが、研究としては仮想空間を用いた遠隔会議の研究も盛んに行われている。しかしこうした研究で開発されるシステムは、参加者の顔画像の表示がされてたりアバターに実名を添えるなど、誰が誰であるか特定できるシステムとなっており、その参加者の判断が求められることに重点が置かれたものとなっている。

そこで、本研究では参加者の情報を特定できない匿名の状態で会議を行うことができるシステムを開発した。匿名で会議を行うことは自らの服装や髪型、顔色や表情などの自らの非言語的情報に注意をする必要がないことや、課題志向的な会議になるなどのメリットが存在するため、本研究ではそれが有効に利用される場面を想定しシステムを開発している。

本システムは、ユーザが没入型ヘッドマウントディスプレイを装着しハンドトラッキングシステムを仮想空間内のアバターの頭部と腕部を一人称視点から直感的に操作することを可能にする。またボイスチャットではリアルタイムに音声処理を施し匿名化するなどの工夫を施し、これらによってユーザの情報を可能な限り隠す。また会議として資料を用いて説明する場面を想定し資料の表示や操作、また相手との資料の共有を行うことが可能である。

# 目 次

<b>第1章 序論</b>	<b>1</b>
1.1 遠隔会議 . . . . .	1
1.2 映像を介した遠隔コミュニケーション . . . . .	1
1.3 仮想空間を介した遠隔コミュニケーション . . . . .	1
1.4 研究の目的 . . . . .	2
1.5 匿名コミュニケーションに焦点を当てる意義 . . . . .	2
1.6 アプローチ . . . . .	2
1.7 本論文の構成 . . . . .	3
<b>第2章 システムの概要とアプリケーションデザイン</b>	<b>4</b>
2.1 概要 . . . . .	4
2.2 システムの構成 . . . . .	5
2.3 アバターの操作 . . . . .	5
2.3.1 頭部の操作 . . . . .	5
2.3.2 腕・手・指の操作 . . . . .	6
2.4 資料の操作 . . . . .	6
2.4.1 ファイルダイアログ . . . . .	6
2.4.2 資料の表示 . . . . .	7
2.4.3 ボタンの各機能 . . . . .	7
2.5 ボイスチャット . . . . .	9
<b>第3章 システムの実装</b>	<b>12</b>
3.1 開発環境 . . . . .	12
3.2 キャラクターアバターの操作 . . . . .	12
3.2.1 キャラクターアバターの詳細 . . . . .	12
3.2.2 アバターの頭部操作 . . . . .	12
3.2.3 アバターの手操作と腕の関節角度の決定 . . . . .	13
アバターの指操作 . . . . .	15
3.2.4 リップシンク . . . . .	16
3.3 資料とインターフェース . . . . .	17
3.3.1 ファイルダイアログ . . . . .	17
3.3.2 資料の変換 . . . . .	17

3.3.3 資料の表示	18
3.3.4 資料の書き込み	18
3.4 ネットワークの動作	20
3.4.1 キャラクターアバター動作の同期	20
3.4.2 ボイスチャット	21
3.4.3 ボイスの匿名化	21
3.4.4 同一の資料画像の表示	21
<b>第4章 操作手法に関する予備実験と考察</b>	<b>23</b>
4.1 実験方法	23
4.2 実験の結果	25
4.3 結果からの考察	26
<b>第5章 関連研究</b>	<b>28</b>
5.1 頭部動作に関する研究	28
5.2 手動作に関する研究	28
5.3 ビデオ会議を改良した遠隔会議システム	29
5.4 仮想空間を用いた遠隔会議システム	29
5.5 没入型仮想空間に関する研究	30
<b>第6章 結論</b>	<b>31</b>
6.1 まとめ	31
6.2 今後の展望と応用	31
謝辞	33
参考文献	34
<b>付録A 実験同意書及びアンケート用紙</b>	<b>37</b>

# 図 目 次

2.1 ユーザの利用想定図	4
2.2 左：アバターが左を向く様子 右：その際のユーザの頭の姿勢	6
2.3 左：ユーザの手と指の形状のアバターへの適用 右：その際のユーザの手と指の姿勢	7
2.4 左上：人差し指で資料を指差しをしている様子 右上：拍手をしている様子 下：拳手をしている様子	8
2.5 左：ファイルダイアログを開くボタン 右：ファイルダイアログ	9
2.6 左：画像ファイルを選択した時に生成される資料 右：PowerPoint ファイルを選択した時に生成される資料	10
2.7 左:書き込み状態に遷移した資料とボタン 右:実際に指によって書き込みを行っている様子	10
2.8 左：正面にあった資料を上下左に移動した図 右：左図のように移動した際に見える映像	10
2.9 発話にあわせてアバターの口を動作させている図	11
3.1 アバターの各ボーンとそれに対応する部位の図解	13
3.2 逆運動学に関する図解	14
3.3 順運動学に関する図解	16
3.4 指の衝突位置の計算	19
3.5 左：衝突位置の画像上での位置の計算 右：1 フレーム後の位置に対して単色の線を引く	20
3.6 資料の共有の図解	22
4.1 上段左:手法1 上段中央:手法2 上段右:手法3 下段はそれぞれの手法に対応した映像の図	24
4.2 タスク完了時間(秒)の各操作手法における平均値	25
6.1 左：仮想空間内にデスクトップを表示させている様子 右：左図においてユーザに見える映像	32

# 第1章 序論

## 1.1 遠隔会議

遠隔会議技術は Egido ら [1] の提唱したもの以来 25 年で大きく進化をしている。背景として、高速なインターネット回線や、ビデオチャットをするのに十分な性能を持ちながらも低コストであるコンピュータが一般に普及したこと、そして技術そのものの普及によるものである。インターネットを介して遠隔で会議をすることで、相手と意思疎通ができるほか、移動に伴う経費や時間の削減に有効であることから実際に会議への代替として用いる事例が増えってきた。

## 1.2 映像を介した遠隔コミュニケーション

今日では、Skype<sup>1</sup> や Meetingplaza<sup>2</sup> といったビデオ会議システムが存在し、個人や企業などで導入されている。こうしたシステムは一般に 1 つの画面に各参加者から送られてきた動画像を分割して表示することにより遠隔地の人同士が一緒に会議を行うものである。こうしたビデオチャット形式の遠隔コミュニケーションは、サービス以外にも様々な研究が存在する。

玉木ら [2] は、映像を介した遠隔会議について、個々の参加者の映像が小さく映像品質に制限があるため、誰がいつ発話し始めるのかを判断しにくく発話が衝突してしまうことが多いことを指摘した。更にこのことから発話の意欲が低下し時間効率が悪くなり、生産性の低い会議になりかねないという問題を指摘していた。そこで人が発話の前に行う特徴的な動作を検知して最も次に発話しそうな参加者を決定し、全参加者へ示すことで話者交替を円滑化する手法を提案しその有用性を示した。Vogler ら [3] は、ビデオチャット形式の遠隔会議において、耳が聞こえにくい、または聞こえないといった聴覚障害者を対象に、いかにして音声による会議の障壁を最小限に抑えるかを主眼に置き、システムを考案した。このように単純なビデオチャットを拡張した様々な遠隔会議の研究が存在する。

## 1.3 仮想空間を介した遠隔コミュニケーション

またこうした様々な遠隔コミュニケーションの研究が存在する中、現在では仮想空間を用いるものも存在する。

---

<sup>1</sup>Skype, <http://www.skype.com/ja/>

<sup>2</sup>Meetingplaza, <http://www.meetingplaza.com/>

浅井ら [4] は、展示会や学会などの相互コミュニケーションを伴うイベントに関して仮想空間上に同様のものを再現し、実空間と仮想空間の間で映像と音声によって相互のコミュニケーションが可能なシステムを開発している。Demeulemeester ら [5] は仮想空間内に円形のテーブルと椅子を設置し、参加者を椅子に座る人型のアバターで表現し、アバターの頭上に参加者の名前を表示することで特定することを可能にしている。この時カメラ画像を用いて実際のユーザの手と腕の動きを検出してそれをアバターに適用することでアバターの操作を行う。これにより拳手やジェスチャーといった簡単な操作が可能であるシステムを開発した。

## 1.4 研究の目的

しかし、こうした仮想空間やアバターを用いた遠隔会議システムの研究では、参加者の個人としての判断が求められるような、参加者の情報に重点が置かれた状況を想定している。つまり参加者が顔を表示したりアバターに実名を添えるなどして誰が誰であるかを特定できるシステムとなっており、実名を出さず匿名で行いたい遠隔会議においての有効な手段ではないという問題が存在する。本研究ではこれを解決しながら匿名性に焦点を当てた遠隔会議を行うシステム開発を目的とする。

## 1.5 匿名コミュニケーションに焦点を当てる意義

インターネットを介した参加者が匿名の遠隔会議は、実際に対面している会議と比べ個人の情報が失われるため課題志向的な会議になると Sproull[6] らは述べている。

また、Walther ら [7] は実際に対面した場合のコミュニケーション(FTF)とコンピュータを介した場合のコミュニケーション(CMC)を比較した。その結果、匿名的な CMC では、相手との親密なコミュニケーション活動が促進され、その結果コミュニケーション相手に対して過度な親密感を感じるという結果を得ている。これは自らの外見や非言語的行動に心的資源を割り当てる必要がないためその分発言内容に多くの資源を割くことができ親密で話しやすい環境を構築できるからであると主張した。

これらから匿名の遠隔会議では例えば、ビジネスの場における会議で会社内で上司や部下などの社会的な関係に関わらず自己の意見を主張し意見を出しあいたい時に有効である。また匿名のいじめ相談窓口において、心を開きながら自分の情報は出さずにいじめに関する情報を提供し意見を仰ぐ時など、自己の情報は伝えず状況のみを相手に伝えたい場合も、匿名であるほうが有効である。

## 1.6 アプローチ

それらをふまえた上で本研究では、匿名の状態に焦点を当てた遠隔会議システムを構築する。ユーザは自らの姿を投影する代わりに仮想空間内でアバターを操作する。またセンサ付きのヘッドマウントディスプレイやハンドトラッキングシステムを用いてジェスチャーを行

うことができる。これは喜多 [12] や Alexandra ら [13] らの、会話の流れを調整する役割や身体動作の同調によって対話が親和的になるといった先行研究の知見によるものである。さらにコミュニケーションとして参加者同士ボイスチャットを行うことができる。しかし声から参加者が特定されてしまう恐れがあるためリアルタイムに音声を加工し特定を防ぐ。さらに会議としての側面を考慮し、資料の提示や共有などの機能を追加した。これらの機能を用いて匿名な状態のまま豊かな表現ができる会議システムの開発を行う。本システムでは匿名で数人で行う小規模な遠隔会議を想定しており、この時各参加者はそれぞれ離れた場所に存在する場合を想定している。

## 1.7 本論文の構成

本論文の構成は、第二章にて作成したシステムの持つ機能と使い方について説明する。第三章ではシステムの実装について述べる。第四章ではシステムの作成にあたり事前に本研究で用いる操作手法が適切なものか検討するために行った予備実験について説明し、その考察とそれに伴う再設計について述べる。第五章では本研究との関連研究について述べる。第六章では本研究のまとめと今後の展望や応用について述べる。

## 第2章 システムの概要とアプリケーションデザイン

本章では、1章で論じた問題を解決するために作成した遠隔会議システムの概要や利用方法、それぞれの機能について説明する..

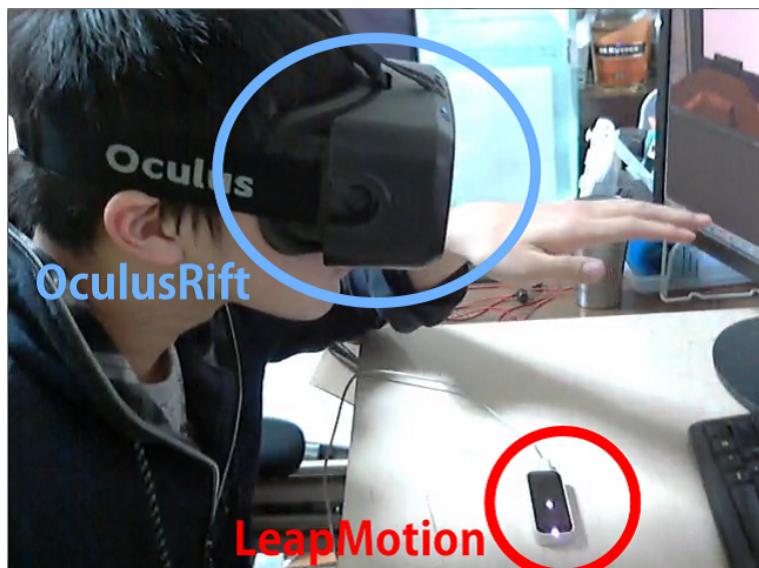


図 2.1: ユーザの利用想定図

### 2.1 概要

本システムの利用において、参加者は、図 2.1 のように椅子に座り、没入型ヘッドマウントディスプレイを装着する。また机の上に置いたハンドトラッキングデバイスに対して手をかざすことで、キャラクターアバターに対して自らの手指の動きと頭の動きを適用し、仮想空間上の会議室にいるキャラクターを一人称視点で直感的に操作することができる。これによって例えば頷きや、両手を用いたジェスチャ、拳手などの意思表示を行うことができる。さらにネットワーク越しに各参加者のアバターの動作を同期することによって相手のジェスチャを認識することができる。

また会議として資料を用いることができる機能が存在する。用いる資料はビットマップなどの画像形式と PowerPoint 形式のファイルを扱うことができる。資料は自らの前に 3 次元の平面物体として描画され、資料を次のページに遷移させたり、資料の画像情報をネットワーク越しの相手に対して共有をするなどの操作を行うなどの利用方法がある。これらの操作はアバターの手で資料の周りに生成されるボタンを押すことによって行うことができる。これらの設計から、本システムはすべての操作においてキーボードやマウスを使う必要がない。これは没入型ヘッドマウントディスプレイの装着によって現実の状況が見えなくなるため、視認して用いるデバイスを使用しにくい問題を考慮したことによるものである。

またコミュニケーションとして参加者同士ボイスチャットを行うことができる。しかし声から参加者が特定されてしまう恐れがあるためリアルタイムに音声を加工し特定を防ぐ工夫を行っている。

本システムが対象とする会議の規模は参加者が 3,4 人の小規模なものを想定したシステムとなっている。またこの時参加者はそれぞれ離れた場所に存在することを想定している。

## 2.2 システムの構成

本システムは Windows7 または 8 が搭載された PC 用のアプリケーションである。使用デバイスはヘッドマウントディスプレイである OculusRift Development Kit 2 とハンドトラッキングシステムである LeapMotion を用いている。

Oculus Rift Development Kit 2 は没入型ヘッドマウントディスプレイであり、ユーザに没入度の高い映像を提供する他、ヘッドマウントディスプレイ内に「3 軸(X-Y-Z) の加速度」「ジャイロスコープ」「地磁気」の 3 種類のセンサーを組み込んでおり、非常に低遅延なヘッドトラッキングを可能にしたものである。

LeapMotion は 2 基の赤外線カメラと赤外線照射 LED で構成された小型 USB 周辺装置である。赤外線 LED に照らされた手や指を 2 機の赤外線カメラで撮影し、画像解析によって 3D 空間での手や指の位置を割り出すことによってユーザのハンドトラッキングを可能にするデバイスである。

## 2.3 アバターの操作

ユーザは自らのアバターの頭部、腕、手、指を操作することができる。またこの各動作はネットワークを介して相手のシステム上でも同様の動作をし、各参加者は相手のジェスチャーの様子を確認する事が可能である。

### 2.3.1 頭部の操作

まず頭部の操作について、OculusRift から得られるユーザの頭のヨー・ピッチ・ロール角からアバターの頭の方向を決定している(図 2.2)。その際アバターの自然な振る舞いを目指すた

め、一定以上の角度が検出されたらそれ以上頭が回転しないようにする処理を施し、代わりに胴体を捻るような処理を施している。またヘッドマウントディスプレイからユーザに対して提供する映像には、アバターの目の位置からの映像を提供することでアバターの視点から一人称視点で操作できるようにしている。

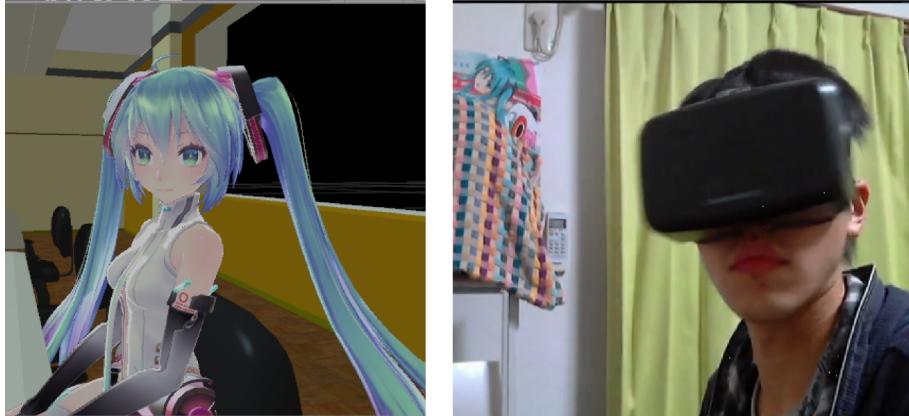


図 2.2: 左：アバターが左を向く様子 右：その際のユーザの頭の姿勢

### 2.3.2 腕・手・指の操作

腕の操作については、図 2.3 のように、机の上に置かれた LeapMotion に対して自身の手をかざすことによって得られる手の位置や姿勢の値を処理しアバターに対して適用している。これにより現実の手の姿勢をそのままアバターで表現することができる。

また指の各関節についても操作することができるため図 2.4 のような指差しが可能であり、その他にも拳手などの豊かなジェスチャ表現が可能である。また拍手などの両手でのジェスチャをすることもでき、このように LeapMotion の検知範囲内においては実際に現実で行われるものと変わらないジェスチャを行うことができる。

## 2.4 資料の操作

仮想空間内で扱うことができる資料の利用方法について説明する。

### 2.4.1 ファイルダイアログ

任意の資料を表示するためにはまずファイルを選択するためのファイルダイアログを開く必要がある。

図 2.5 左がシステム開始時にユーザに見えている映像である。そしてこれに表示されている円形のボタンを押すことによってファイルダイアログを開くことができる。図 2.5 右はファイ

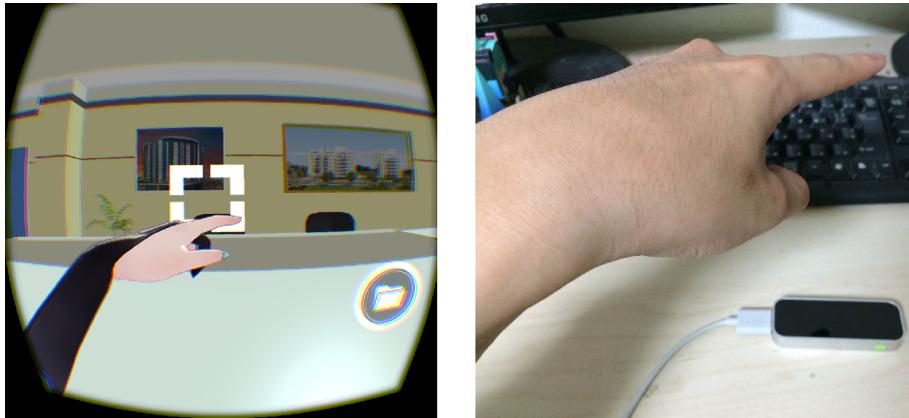


図 2.3: 左 : ユーザの手と指の形状のアバターへの適用 右 : その際のユーザの手と指の姿勢

ルダイアログを開いた際のユーザに見えている映像であり、各ファイルの右に表示されている円形のボタンを押すことによって操作する。

#### 2.4.2 資料の表示

こうして選択された資料は図 2.6 のようにアバターの目の前に生成される。図 2.6 の左右の違いは、画像ファイルを元にした資料では 1 ページのみであるため次へ進む、前に戻る機能のボタンが存在しないが。PowerPoint ファイルは資料が複数にわたっているためそれが存在する。また資料を開いたままファイルダイアログからファイルを再度選択することによって、同時に複数の資料を表示させることができる。

#### 2.4.3 ボタンの各機能

表 2.1 に資料に付随するボタンの各機能を示した。

この表において特に解説が必要である書き込み、共有、移動ボタンの機能について説明する。

まず資料への書き込みを行うことができる状態へ移行する機能について、図 2.7 に示す。図 2.7 左は書き込みをすることが可能な状態に遷移した資料の図である。同時に生成されるボタンは右が書き込みを削除して書き込み状態を解除するボタン、左上が書き込みを元の資料に適用するボタン、左下がローカルに書き込んだ画像を保存するボタンである。

ユーザは図 2.7 右のようにアバターの指によって資料に書き込みを行うことができる。このように書き込んだ資料画像を共有ボタンと組み合わせ、お互いに資料の書き込みを共有することも可能である。

資料をネットワークを介して相手に共有する機能では、仮想空間内の大画面に対して資料を表示する。同時に相手のシステム上の大画面にも同じ画像を表示させる。これにより同時に同じ資料を共有することができる。またアバターの正面ではなく前方の大画面に対して表



図 2.4: 左上：人差し指で資料を指差しをしている様子 右上：拍手をしている様子 下：挙手をしている様子

示すことによって、相手が共有した大画面を見ているかどうか判断することが可能である。これは各ユーザのアバターの頭部動作が同期されていることによるものである。

資料を移動させるボタンでは、ボタンを押すと指の先端に資料が追従する。また一定時間指をその場に留めると資料を追従を終了し、その場に資料を固定する機能を持つ。これにより正面に座る相手を確認しながら複数の資料を確認することができる。これは常にカメラに対して正面を向くように回転するため、図 2.8 のように下に移動させて机に資料がめり込んで一部が視認できなくなってしまうこと、上に移動させてボタンが押しにくくなってしまうこと、横に移動して資料が読みにくくなることを防いでいる。



図 2.5: 左：ファイルダイアログを開くボタン 右：ファイルダイアログ

書き込み	共有	移動	閉じる	次に進む	前に戻る
資料への書き込みを行うことができる状態へ移行する	資料の画像情報をネットワークを介して相手に共有する	資料を移動させる	資料を仮想空間上から消す	資料の画像情報を次のページのものに変更する	資料の画像情報を前のページのものに変更する

表 2.1: 各ボタンの機能

## 2.5 ボイスチャット

本システムではボイスチャットによるコミュニケーションを可能としている。しかし声を加工することなくボイスチャットをすると声からユーザが誰であるか特定されてしまう問題がある。そのため本システムではリアルタイムに音程のピッチを上げる処理をすることによって特定されにくくしている。

また図 2.9 のように、発話に合わせてアバターの口を開ける処理を行なったかもアバターがしゃべっているかのように口を動かす処理を行なっている。またこの処理は頭や手などと同様にネットワークを介してユーザ間で同期されるため、これにより頭部操作と合わせて誰が誰に向かって発話しているかを映像から判断することが可能となる。

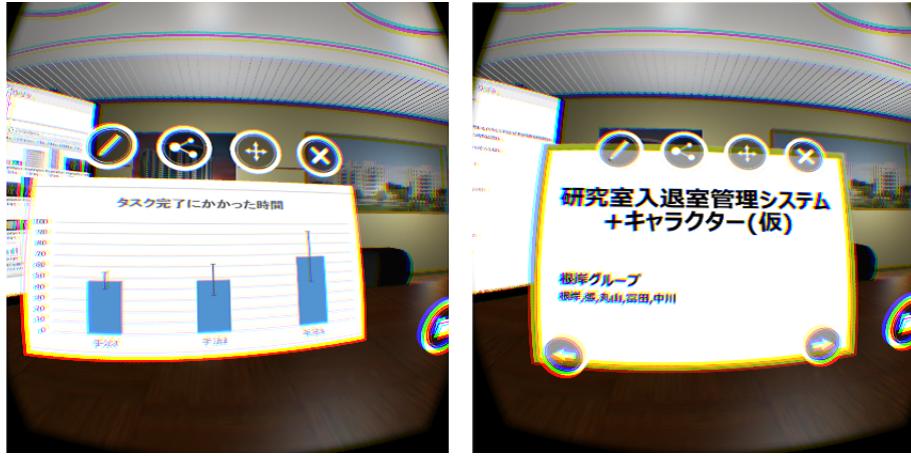


図 2.6: 左:画像ファイルを選択した時に生成される資料 右:PowerPoint ファイルを選択した時に生成される資料



図 2.7: 左:書き込み状態に遷移した資料とボタン 右:実際に指によって書き込みを行っている様子



図 2.8: 左:正面にあった資料を上下左に移動した図 右:左図のように移動した際に見える映像



図 2.9: 発話にあわせてアバターの口を動作させている図

# 第3章 システムの実装

本章では、本システムのそれぞれの機能を実現するための実装手法について説明する。

## 3.1 開発環境

本システムのソフトウェアの実装はWindows7が搭載されたPCでUnity5.0.1fとC#で行われている。各デバイスのソフトウェアへの組み込みには各々のUnity用SDKを利用して実装されており、OculusRiftのSDKにはOVR SDK0.5.0.1を使用し、LeapMotionのSDKにはLeapMotion SDK ver2.2.7+30199を用いている。

これらはUnity用SDKが配布されており、これらデバイスのシステムへの組み込みにはそれぞれのUnity用SDKを用いた。また、ネットワークによる各キャラクターの動作の管理やボイスチャットにはPhotonUnityNetworkライブラリを用いた。これも同様にUnity用のSDKが配布されており、それをシステムに用いた。

## 3.2 キャラクターアバターの操作

この章では頭部と腕部、手指部の操作の実装、またリップシンクの実装についてユーザがアバターを操作する方法の実装部分とについて述べる。

### 3.2.1 キャラクターアバターの詳細

今回本システムで用いる人型のキャラクターアバターはボーンが細分化されているものを用いた。その理由としては図3.1のように首関節や、腕関節、腰関節、足関節などをそれぞれ別のボーンとして扱うことができ、細かい首や腕、手の動きなどを表現することができるためである。

### 3.2.2 アバターの頭部操作

キャラクターアバターの頭部の動作であるが、これはOculus Riftを装着したユーザの頭部の方向と同期している。これはOculusRiftに組み込まれた3つのセンサーによって、ユーザの頭部のヨー・ピッチ・ロール角を取得し、SDKによってそれぞれX,Y,Z軸の回転角度の数

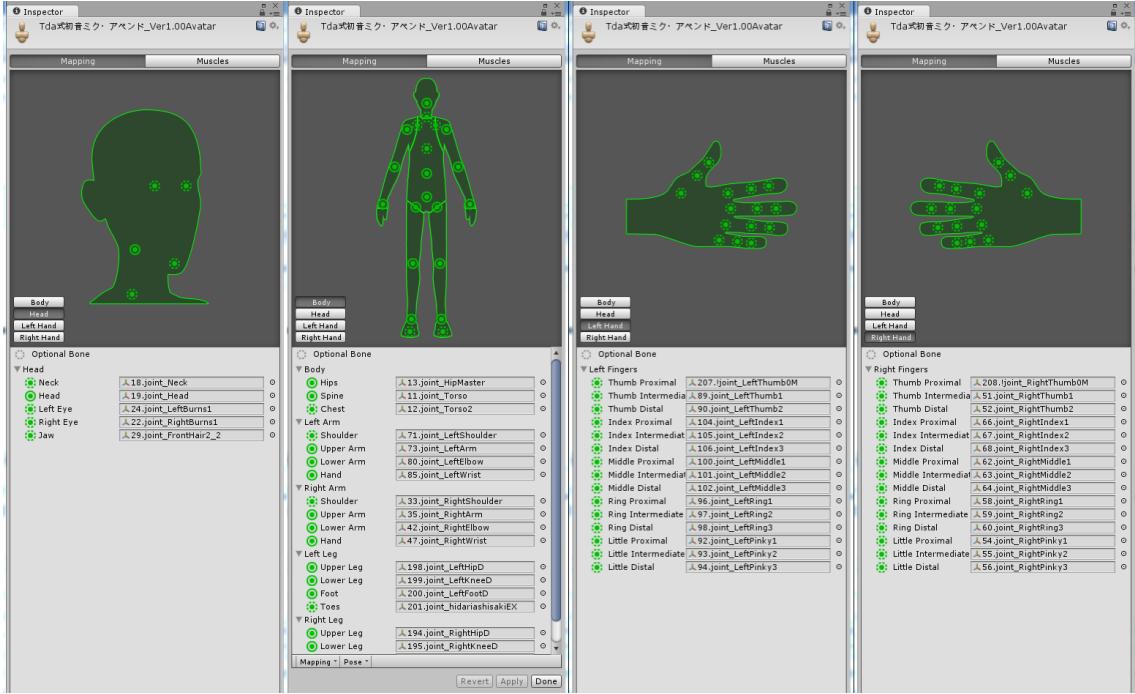


図 3.1: アバターの各ボーンとそれに対応する部位の図解

値を取得することができる。これを図 3.1 の Neck ボーンに対して適用することで、頭部方向の同期を行っている。

またこの時左右は良いが、上下を向いた時に Neck ボーンのみで表現すると顎が引いたような形となり不自然であることから、ピッチ角が変化した場合ある程度 Chest ボーン、Spine ボーンにも角度をつけることによって後ろに少しのけぞるようにし、より自然なアバター動作を表現している。

### 3.2.3 アバターの手操作と腕の関節角度の決定

キャラクターの手の操作について、LeapMotion では LeapMotion に対して手をかざしたユーザの手の位置を検出している。この手の位置をそのままアバターの手の位置とし、適用している。

しかしこの時、キャラクターアバターについて、適用した手の位置に対して適当な腕関節を決定しなければならないという問題が発生する。この問題に対処するために、手の位置とそれに基づいた逆運動学処理を利用し、キャラクターの腕関節角度を決定している。

逆運動学とは、人型アバターでは関節の末端部分の位置は常にその親となる関節の位置と角度に依存することを利用し、末端部分の位置を先に決定し、その位置を実現するための親となる関節の角度を逆計算する手法である。今回は LeapMotion によって手の位置が取得でき

ているため、それを利用しキャラクターアバターの関節角度を逆計算することによって適当な関節角度を決定することを考える。

以下に計算方法を述べる。

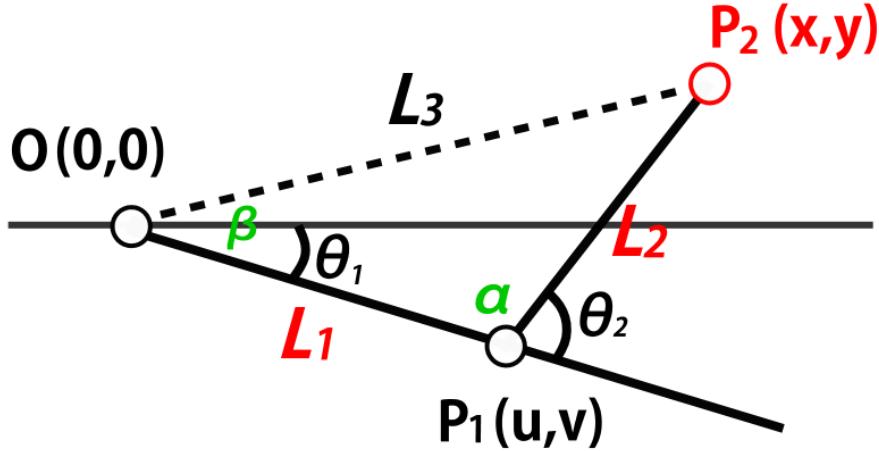


図 3.2: 逆運動学に関する図解

腕を簡易にあらわしたものが図 3.2 であり、ここで求めるべきは  $\theta_1$  と  $\theta_2$  である。図 3.2において、原点 O は肩関節を表しており、 $P_1$  は肘関節を表し、 $P_2$  は手関節を表している。この時、原点 O の座標と、赤字で示されている手の位置を表す点  $P_2$  の座標と上腕の長さを表す  $L_1$  と前腕の長さを表す  $L_2$  は既知である。

また  $OP_2$  の長さを表す  $L_3$  も O と  $P_2$  より  $\sqrt{x^2 + y^2}$  と求めることができる。

まず  $\angle \alpha = \angle OP_1P_2$  とし、 $\angle \beta = \angle P_1OP_2$  とする。

この時余弦定理から、

$$\cos \alpha = \frac{-(x^2 + y^2) + L_1^2 + L_2^2}{2L_1L_2} \quad (3.1)$$

と求めることができる。

よって、式 3.1 より

$$\alpha = \cos^{-1}\left(\frac{-(x^2 + y^2) + L_1^2 + L_2^2}{2L_1L_2}\right) \quad (3.2)$$

となる。

次に  $\beta$  について余弦定理より

$$\cos \beta = \frac{-L_2^2 + L_1^2 + (x^2 + y^2)}{2L_1\sqrt{(x^2 + y^2)}} \quad (3.3)$$

と求めることができる。よって式3.3より

$$\beta = \cos^{-1}\left(\frac{-L_2^2 + L_1^2 + (x^2 + y^2)}{2L_1\sqrt{(x^2 + y^2)}}\right) \quad (3.4)$$

となる。

以上より、図3.2と式3.2、式3.4から

$$\theta_1 = \tan^{-1}\left(\frac{y}{x}\right) \pm \beta \quad (3.5)$$

$$\theta_2 = \pm (\pi - \alpha) \quad (3.6)$$

で求めることができる。

このように、リアルタイムに手の位置情報が更新され、そこから随時計算することにより、式3.5と式3.6のように各関節の角度を決定することができる。それをアバターの肩関節と肘関節に対して適用している。

この時仮にLeapMotionで手を検出していないまたはできなかった場合には上述した逆運動学による計算は必要ないので、キャラクターアバターの動作は椅子に座りながら膝に手を置いているといったものになる。

### アバターの指操作

キャラクターの指の操作について、LeapMotionではLeapMotionに対して手をかざしたユーザの指の関節角度を値として取得することができる。具体的には10本の指のMP関節、PIP関節、DIP関節を取得することができ、それをアバターの両手の計10本の指に対してそれぞれ適用している。

腕の関節角度の計算とは違い、指に関しては既に各関節の角度がわかっているので、逆計算をすることは不要である。よって、順運動学にもとづいてアバターの各関節に対して角度を適用している。

順運動学とは、与えられた関節の角度から末端の位置を求める方法である。

以下に計算方法を述べる。

図3.3において、MP関節を原点Oとし、PIP関節を $P_1$ 、DIP関節を $P_2$ 、指先を $P_3$ と定義する。

図3.3であるがここで求めるべきは、 $P_1$ と $P_2$ と $P_3$ の座標である。これがわかれることによつて指の姿勢を一意に決めることができるからである。ここで、赤字で示されているもの、指の長さである $L_1$ と $L_2$ と $L_3$ とMP関節角度 $\theta_1$ 、PIP関節角度 $\theta_2$ 、DIP関節角度 $\theta_3$ は既知である。

まず原点Oの座標が(0,0)であることから $P_1$ の座標 $(u, v)$ は

$$u = L_1 \cos \theta_1 \quad (3.7)$$

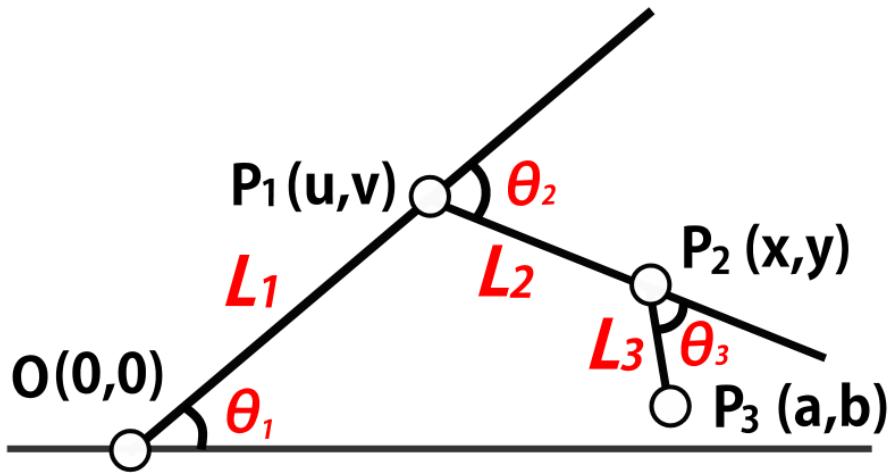


図 3.3: 順運動学に関する図解

$$v = L_1 \sin \theta_1 \quad (3.8)$$

と求めることができる。式 3.7, 式 3.8 を用いて,  $P_2$  の座標  $(x, y)$  を求めることを考える。

$$x = u + L_2 \cos(\theta_1 - \theta_2) \quad (3.9)$$

$$y = v + L_2 \sin(\theta_1 - \theta_2) \quad (3.10)$$

のように、求める事ができる。同様に式 3.9, 式 3.10 から  $P_3$  の座標  $(a, b)$  を求めると,

$$a = u + x + L_3 \cos(\theta_1 - \theta_2 - \theta_3) \quad (3.11)$$

$$b = v + y + L_3 \sin(\theta_1 - \theta_2 - \theta_3) \quad (3.12)$$

以上の計算より、既知の情報から各関節の位置を求めることができる。これらをアバターの指に対して適用することで、LeapMotion によって検出された指の姿勢をアバターで表現することができる。

### 3.2.4 リップシンク

ユーザの発話に伴いアバターの口が動く機能を実装している。この実装はマイクから取得する発話の音量の値によって口部分の形状を変更する実装を行っており、ユーザの発音に伴つて口が動くという厳密なリップシンクではない。

口部分が最大まで開いた場合の開き度を 1, 閉じている場合を 0 と設定している。またマイクから取得できる音量はシステム上では 0 から 1 の間で値で扱われる。このことを利用し、取得できる音量を口の開き度に対して値を入力することによってリップシンクを可能にしている。

しかしこの実装では音量のみにフォーカスしている。本システムが想定する環境では参加者はそれぞれ離れた場所にいるためそれぞれの声が自らのシステムの音量として取得されることはない。しかし声ではなく例えば大きな物音が発生した場合でも誤作動を起こしてしまう問題がある。そのため、四倉ら [9] のように、予めカメラによって撮影された複数の表情画像と音声特徴量をひも付け、音声からリアルタイムに推定してアバターの表情を変える手法を用いるなど工夫が必要である。これは今後の課題とする。

### 3.3 資料とインターフェース

この章では資料の表示処理の実装や、インターフェースの操作の実装について述べる。

#### 3.3.1 ファイルダイアログ

図 2.5 に示すファイルダイアログは、資料として提示できるファイルをリスト化して表示するものである。ここで列挙されるファイルはシステムの実行ファイル以下に存在する StreamingAssets フォルダに存在するもの、かつ、ファイルの拡張子が.ppt,.pptx,.jpg,.bmp,.png のものである。

本システムは Unity を用いて作られているがビルド時に必要なアセットを結合し実行ファイルを出力するため、本来であればユーザが資料を追加したい場合はその都度ファイルを追加して再度ビルドしなければならない。しかしふルド時に StreamingAssets フォルダ以下のファイルは結合されることはない。このことを利用して、ユーザは資料のファイルを StreamingAssets パス以下に置くだけでシステム内で扱える資料を追加することができ、システムはファイルパスへの参照によりそれらを扱うことができる。

#### 3.3.2 資料の変換

本システムでは、仮想空間上に用いることができる資料の形式として PowerPoint ファイルと画像ファイルがある。先ほど述べたファイルダイアログで PowerPoint ファイルを選択すると、選択直後に資料をページごとに一枚ずつ画像化する処理が始まり最終的にページ数分の画像が生成される。

資料をユーザの前に提示する際 PowerPoint ファイルをそのままのファイル形式で本システム上で使用することはできない。そのため資料をページごとに画像に変換する必要があるためこのような処理が行われている。これを実現するために、PowerPoint ファイルを画像に変

換する処理を行う自作の実行ファイルを StreamingAssets フォルダ以下に置いている。ファイルの設置場所の理由は先ほど述べた資料のファイルの理由と同様である。

この実行ファイルには PowerPoint ファイルのファイルパスと生成される画像の縦の長さ、横の長さ、出力先のファイルパスを与えるものであり、ファイルパスから PowerPoint ファイルを参照し、与えられた縦の長さ横の長さを元に画像を生成し、最後に与えられた出力先のファイルパスを元にフォルダの生成とそのフォルダ以下への出力を行う。現在の実装では、第一引数はユーザが選択することができるが、他のパラメータはシステムが決定をしており変更することはできない。そのパラメータは、それぞれ 640, 480, StreamingAssets / ファイル名\_Images / となっている。このことより現在では PowerPoint の縦横比が 4:3 のものに対しては最適な変換を行うことができる。

また変換後のファイルのサイズは文字だけのページなら 60KB 程度、画像が含まれたページでは 200KB となっており、以下で説明するネットワーク上の資料画像の共有において通信帯域を極端に圧迫する可能性は低い。

### 3.3.3 資料の表示

ファイルダイアログで選択されたファイルは、変換処理後から自動で図 2.6 のようにユーザの前に生成される。

### 3.3.4 資料の書き込み

図 2.7 にあるように、本システムでは指による資料への書き込みを行うことができる。これの実現には、3 つの処理が行われる必要がある。それぞれの処理を図示したものを図 3.4 と図 3.5 に示す

ここで図 3.4 および図 3.5 は、仮想空間を x,y,z 軸で表したものである。また、ピンク色の矩形は仮想空間内で表示されている資料を表しており、灰色の直方体はアバターの指の衝突判定を表している。

まず、図 3.4 に示すように資料に対する仮想空間上での指の衝突位置の計算を行う。指の衝突判定は直方体になっているため資料との衝突は点ではなく面である。しかし計算の便宜上衝突面の角 4 点を衝突点として計算で扱う。この 4 点の座標は取得することができるので、そこから座標の平均値である  $(x_v, y_v, z_v)$  を計算する。

これは

$$x_v = \frac{(x_{v1} + x_{v2} + x_{v3} + x_{v4})}{4} \quad (3.13)$$

$$y_v = \frac{(y_{v1} + y_{v2} + y_{v3} + y_{v4})}{4} \quad (3.14)$$

$$z_v = \frac{(z_{v1} + z_{v2} + z_{v3} + z_{v4})}{4} \quad (3.15)$$

と計算することができる。

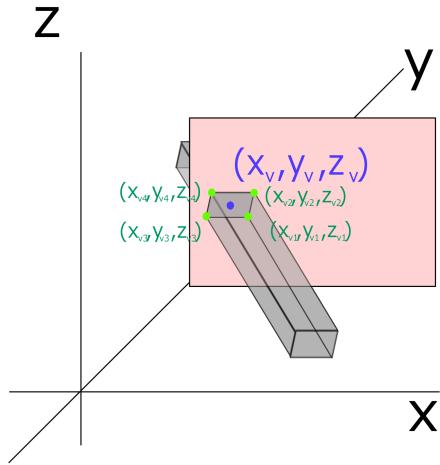


図 3.4: 指の衝突位置の計算

次にこうして計算した  $(x_v, y_v, z_v)$  について、衝突位置を画像上の位置に変換する処理を行う。図 3.5 左に示す。 $(x_v, y_v, z_v)$  はワールド座標系で扱われる座標であり、それを資料平面上のローカル座標として扱えるよう処理を行うことで資料上の座標  $(x_i, y_i)$  を得る。

以上の処理を 1 フレームごとに行っており、その都度  $(x_i, y_i)$  を得ている。最後に、図 3.5 右に示すように、あるフレームでの資料上の座標を  $(x_{i1}, y_{i1})$  とし、その 1 フレーム後の座標を  $(x_{i2}, y_{i2})$  とする。これら 2 点間にに対して単色の線を描くことによって資料に対して書き込みを行うことを達成している。

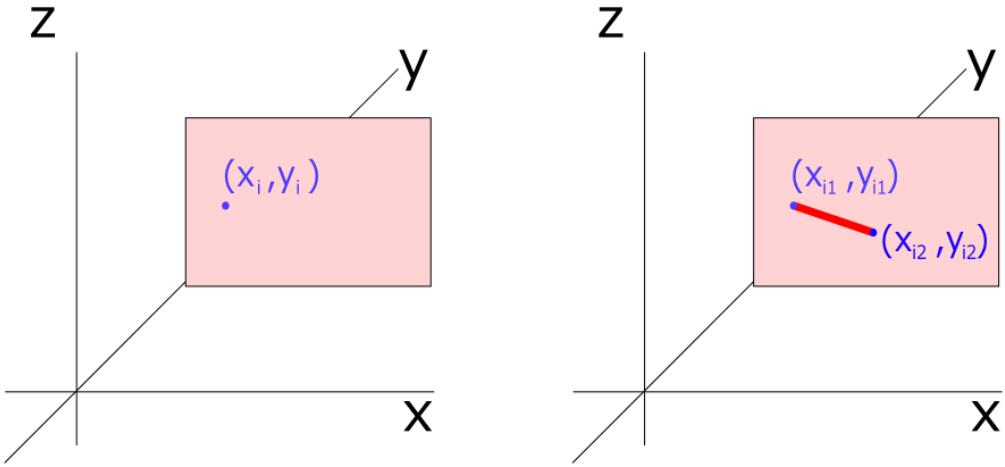


図 3.5: 左 : 衝突位置の画像上での位置の計算 右 : 1 フレーム後の位置に対して单色の線を引く

### 3.4 ネットワークの動作

この章では仮想空間での遠隔会議を行うためのネットワーク機能について説明する。本システムにおいてネットワークを活用する機能は 3 種類あり、各参加者によるキャラクターアバターの動作の同期、ボイスチャット、資料などの画像ファイルの同期である。

#### 3.4.1 キャラクターアバター動作の同期

本システムのキャラクターアバター動作の同期に関しては PhotonUnityNetwork というネットワーク処理用ライブラリを用いて実装を行っている。各キャラクターごとに、図 3.1 で示したものの中で、頭部同期による変更がなされる Neck, Spine, Chest ボーン、両手同期による変更がなされる Left と Right 両方の Shoulder, UpperArm, LowerArm, Hand ボーンと、指の同期による変更がなされる Left と Right の両方の ThumbProximal Intermediate Distal, IndexProximal Intermediate Distal, MiddleProximal Intermediate Distal, RingProximal Intermediate Distal, LittleProximal Intermediate Distal ボーンが変更の対象である。これらボーンについて、3 軸の回転角度がそれぞれリアルタイムに取得することが可能である。本システムではそれぞれの回転角度を取得し、それを PhotonUnityNetwork によって Photon サーバに値を送信し、それをユーザ間で同期させることによってキャラクターアバターの動作の同期を達成している。

この時それぞれのボーンの位置を取得しない理由は、アバターのボーンの親子構造により、回転角度のみを同期させることで本システムで想定される全ての動作の表現が可能だからである。また位置を取得しないことにより、同期に必要な通信量の削減も考慮している。また

このアバターのボーンの親子構造はキャラクターアバター毎に異なっているが、本システムでは事前に正常な動作となるように調整をしているため構造の違いによる問題は発生しない。

また動作の同期において、特に手を振るなど激しい動きをした場合そのまま通信によって得た回転角度を適用すると、動作がカクつきキャラクターアバターの表現が不自然になる可能性がある。そのため各ボーンの角度について現在の角度と通信によって得た角度間で線形補間を行うことでなめらかな動きになるよう処理を加えている。

### 3.4.2 ボイスチャット

ボイスチャットでは、従来の遠隔会議と同様にマイクからユーザの音声を取得し、それを共有することで行う。その際発言者本人が声から特定されるのを防ぐものである。

各参加者の声の同期について説明する。まず、マイクからユーザの声を取得し、その音声情報を通信の帯域幅を考慮し圧縮してエンコードする。そのエンコードが終了したことを検知し、エンコード済みの音声情報のパケットを自分を除く他の参加者に対して送信する。ボイスの受信者側はそれを受け取り、再生する。以上の動作によってボイスの同期を可能にする。

### 3.4.3 ボイスの匿名化

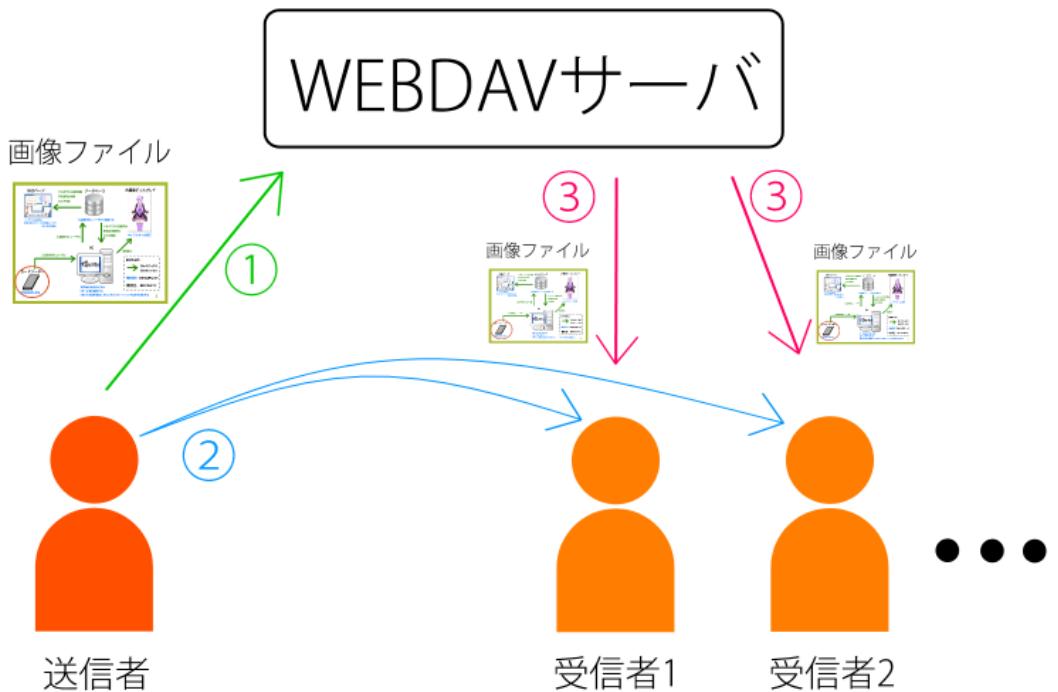
発言者本人が声から特定されるのを防ぐ匿名化処理について説明する。ボイスの同期について、送信されたボイスを受信して再生する時、ボイスは他のユーザにとってはシステムの仮想空間内に存在する各アバターから再生されているものとして扱われる。これを利用しほぼ音が再生される前に、Unity の AudioMixer 機能を用いてリアルタイムに音声情報を加工することによって匿名化を実装する。

本システムでは PitchShifter と呼ばれるミキシング手法を用いて、ボイスのピッチを変更し、匿名化を可能にしている。

### 3.4.4 同一の資料画像の表示

本システムでは、前方の大画面に対して各ユーザ間において同一の画像を表示する機能を持っている。参加者に対して直接ファイルを送信することができないため、この機能を実現するためにファイルサーバと PhotonRemoteProcedureCall を用いて擬似的に同一の資料画像を共有し、各ユーザによって同一の画像を表示させる事を考える。図 3.6 に、動作を示す。

まず全員に資料を配布したい人間を送信者とし、他の参加者を受信者とする。送信者側のシステムの動作は、前章で行われた画像化された PowerPoint によって作成された資料の各画像について、変換と同時にシステムによって変換後のファイルを非同期処理を用いてサーバに自動的にアップロードするものである。これは.png などの画像ファイルが共有するファイルとして選択された場合も同様にアップロードが行われる。



- ① 資料を画像化し、WEBDAVサーバへ資料画像のアップロード
- ② PhotonRPCによりサーバから画像をダウンロードするよう各ユーザに命令を送信
- ③ 指定の画像をダウンロードし、前方の大画面に表示させる。

図 3.6: 資料の共有の図解

そして全てのファイルのアップロードが完了した後を考える。送信者側が資料を共有するボタンをクリックした時、送信者側のシステムが受信者側のシステムに対して PhotonRemoteProcedureCall を用いて、サーバから画像をダウンロードするように命令を送信する。受信者側のシステムはその命令の元、サーバから画像をダウンロードする。ダウンロードが完了した後、その画像を前方の大画面に対して適用する。

このように同一画像の表示に必要な動作はシステムによって自動的に行われるものであり、共有したい資料を選択する以外にユーザが行わなければならない操作はない。以上の動作によって送信者と受信者によって同一画像を表示することを可能にする。

# 第4章 操作手法に関する予備実験と考察

本章ではシステムの開発にあたり資料の操作手法が有用性と使用感を確かめるために行つた予備実験とその考察について述べる。

## 4.1 実験方法

本システムは没入型ヘッドマウントディスプレイを用いるため現実のデバイスなどを視認することはできない。そのため操作において視認の必要があるキーボードやマウスなどの入力デバイスを用いることは不適当だと考えた。またハンドトラッキングシステムによってジェスチャを行うため入力デバイスを持つことはそれを阻害してしまうと考え 3 種類の手法に絞り込んだ。

図 4.1 に示したような 3 種類の操作手法を用意した。手法 1 は、OculusRift をタップすることによってクリックとみなすものである。映像では正面に照準を表示させその位置にあるボタンにクリック処理を施す。手法 2 は机の上に LeapMotion を置きそれに手をかざすことで仮想空間内に手を出現させる。この手の人差し指には棒状のオブジェクトがついておりこれをカーソルとする。図 4.1 の下段中央のようにボタンにそれを一定時間重ねているとクリックとみなす。手法 2 は現時点で本システムで用いられている手法である。手法 3 は OculusRift の前面に LeapMotion をとりつけて正面に手をかざすことで仮想空間内に手を出現させる。手の役割やクリック方法は手法 2 と同様である。

以上 3 種類の方法を用いてそれぞれその操作の正確性や使用感を確かめるため実験を行つた。被験者 9 人(男性 7 名女性 2 名)にタスクを行つてもらった。被験者には一度これら 3 種の手法をそれぞれ練習してから仮想空間内に存在する 10 個の仮想空間のメニューを選択し規定の位置へと操作するというタスクを行つてもらった。そのタスク完了時間を計測し、操作に正確性を求めた場合の早さを確かめる。またタスクの完了後被験者には使用感に関する感想をアンケートに書いてもらい、実際の使用感を確かめた。

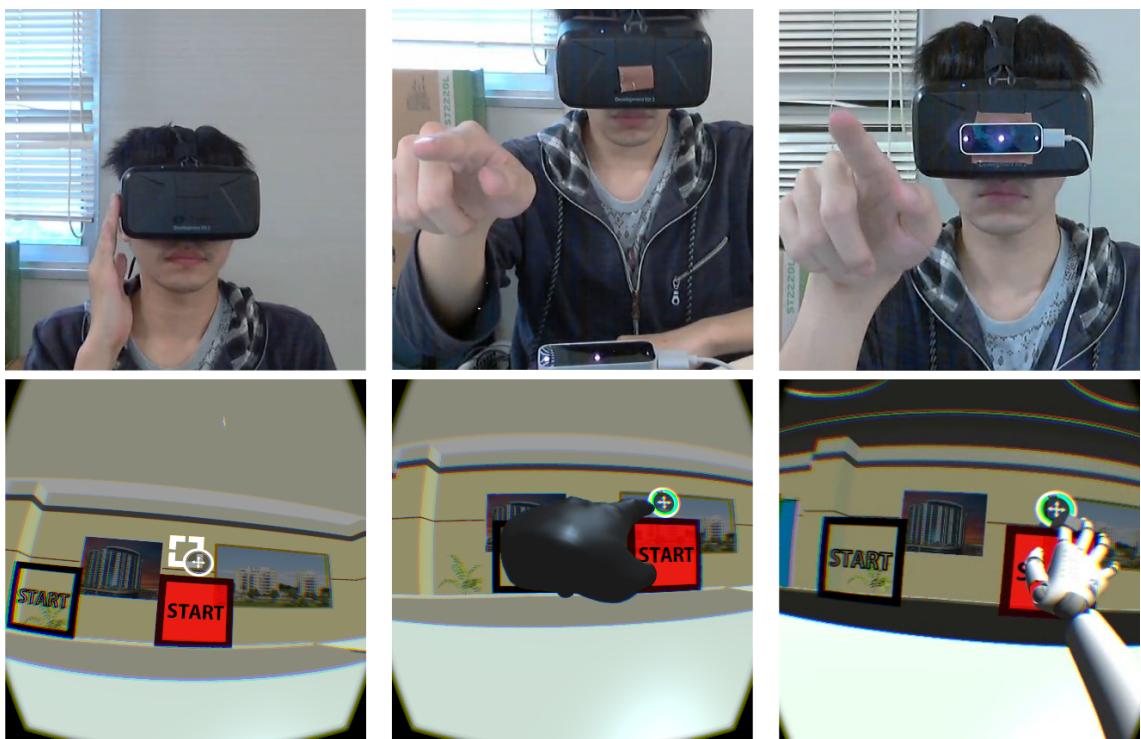


図 4.1: 上段左:手法1 上段中央:手法2 上段右:手法3 下段はそれぞれの手法に対応した映像の図

## 4.2 実験の結果

図 4.2 はそのタスクの完了時間をグラフ化したものである。

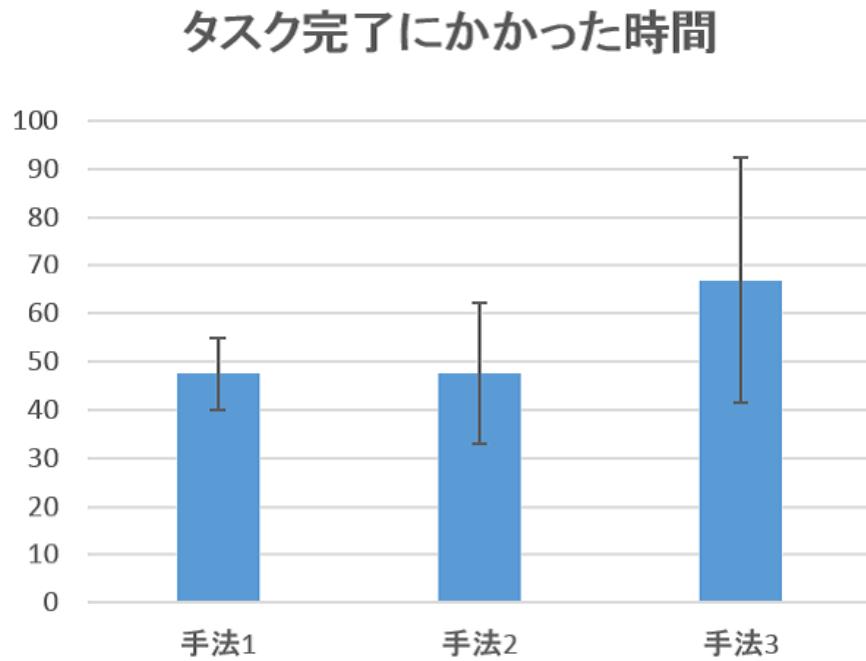


図 4.2: タスク完了時間(秒)の各操作手法における平均値

被験者 9 人から得られた実験結果として、タスクの完了時間の平均は手法 1 が 47.454 秒、手法 2 が 47.589 秒、手法 3 が 66.907 秒であった。またタスクを完遂できなかった被験者はいなかった。平均の結果を見ると手法 1 と手法 2 の早さが等しく手法 3 が遅いように思えるが、得られたデータに対して二群の平均値に差があるか t 検定を行った所、手法 1 と 3 では有意傾向ではあるが有意差は認められず ( $t(16)=2.07, p=0.054 > 0.05$ )、手法 2 と 3 に関しても同様であった ( $t(16)=1.867, p=0.085 > 0.05$ )。

使用感の感想については、まず手法 1 では力加減がわからない、頭ごと目標に向けなければいけないため操作が難しかった、などのネガティブな意見が多く見られた。実際に練習の際、実験で用いるシステムを制作していた際に想定していたよりも弱くタップしていた被験者が多かった。手法 2 では、ポジティブな意見として手を用いた操作は直感的であった、精度がよく操作がしやすかった、奥行きを操作できるのは良かったという意見を得ることができた。ただし一方で操作をする際に腕が疲れるというネガティブな意見も得た。手法 3 についてはポジティブな意見として手法 2 よりも操作が直感的であったという意見を得た。一方で精度が悪かった、操作に腕が疲れるといったネガティブな意見も得た。

### 4.3 結果からの考察

平均達成時間が遅いが有意差が認められなかったことについて、手法3ではトラッキングの精度に問題があったためと考える。LeapMotionによる手のトラッキングは赤外線画像から指を認識して行うものである。しかしこの時正面に対して指を向けるという設計上、図4.1の上段右のようにLeapMotionに対して指の方向が平行になり、指が手の甲によって隠れやすくなってしまう。これにより結果としてトラッキング精度が落ちてしまう問題があった。一方手法2は、下から赤外線を照射しそれに対して手をかざしているため正面にシステムの設計上5本の指を認識しやすく精度が高かったと考えている。

実験において指が手の甲に隠れてしまう問題に気づいた4人の被験者が手をLeapMotionに対して平行にならないようにすることで操作をスムーズに終えており、手の方向が平行になっているまま操作した被験者は精度の低下により操作の完了までに時間がかかってしまっていた。これが平均値には差があるが、有意差が認められなかった原因であると考える。手法3のこの問題を解決するにはアバターの手のひらにカーソルを手に垂直になるように設置することで、手のひらで押し出すようにクリックするという操作手法に変えることによりLeapMotionに対して手の向きを垂直にして精度をなるべく高めながら操作を行う方法が考えられる。しかし実験中全員が正面に指を差す手の姿勢をとっていたため精度のために手法3よりも手法2のほうが、精度よく直感的な操作を行うことができると考察した。

使用感の感想について、手法1では操作のしにくさの感想が目立った。操作がしにくいうがこのことタイムに影響していない理由としては、タップという操作は再度行う操作をしても時間がかかるので多少のタップミスがあっても影響しなかったと考える。また想定していたよりもタップが弱い被験者が多く見られたことについては、改善案の1つとしてタップの検知条件をゆるくすることが考えられる。しかしこの処理はヘッドマウントディスプレイの加速度センサの値を用いているため首振り動作でもタップと誤検知してしまう場合がある。さらに直感的な操作であるといった意見が得られなかったことも含め、この手法の採用は見送った。

また手法2,3について共通して直感的な操作ができた、前後の奥行きの操作ができるることはよかったです、などのポジティブな意見が見られた一方で腕が疲れたという意見が多数見られた。これはシステムで一定時間重ねているとクリックとみなすという手法を用いているからであると考えられる。中村ら[8]は現実の大画面環境における研究で、ポインタを一定時間静止する操作をウェイティングと呼称しており、認識のブレにより位置がずれてしまう問題や決まった時間だけ静止しないなければならないため選択に時間がかかるなどの問題が起きる可能性を指摘していた。また、これを解決するためにダブルクロッシングという対象を短時間に2度横切ることでクリックとみなす手法を提案していた。本研究でも腕の疲れという問題を解決するためにダブルクロッシングが有効である可能性があると考えた。

しかし中村らの研究と違い本研究では手の操作の役割として資料の操作だけではなくジェスチャを行うことも想定しているのでジェスチャの際に資料の誤操作が起きてしまう可能性もある。そのため本システムではウェイティングの手法を採用した。

これらから本システムでは手法2を採用した。この時生じる腕が疲れる問題には必要なウェ

イティング時間を短くするように再度実装し対処した。

## 第5章 関連研究

### 5.1 頭部動作に関する研究

石井ら [10] は、1つの画面に各参加者のアバターを分割して表示するシステムで、3次元CGアバタは表情や視線、身振りなどの身体動作を中心としたノンバーバル情報が欠如するため、その補完が仮想空間での円滑なコミュニケーションの促進に重要なものと考え、特にノンバーバル情報の中でも、会話中の視線に焦点を当て、視線の会話の開始合図や発話権の授受発話の促進などの重要な機能に着目し、システムを開発した。このシステムは受話者が発話者に対して、話者は直前に発話していた受話者に対して視線を向ける傾向が強いことを利用し、話者注視効果と発話開始効果から算出した最も注目度が高いユーザアバタに対して視線を向けるものである。また評価実験から発話意欲促進に有効性があることを確認している。

Vertegaal ら [11] はこうした発話中の他者からの視線による影響について述べており、発話中に他者からの視線を感じた時には発話が促進されるということを実験によって確認した。

更に、井上ら [14] は、コミュニケーション時の頷き動作に着目し、コンテキストの同調の度合いに伴って頷き動作が増加していることを実験によって確認した。

高橋ら [15] は時系列信号処理を用いて、カメラから取得した顔の特徴点を元にリアルタイムにアバターの頭部姿勢や、表情などを生成する手法を開発した。本研究ではユーザはヘッドマウントディスプレイを装着しているため表情を認識することはできない。しかし、本システムでの今後の課題の一つとして厳密なリップシンクを挙げたが、高橋らの手法を用いて口の特徴点をカメラから取得しそこから得られた口形状をアバターの口形状への入力することでリップシンクを達成できる可能性がある。

### 5.2 手動作に関する研究

喜多 [12] は対話におけるジェスチャーの役割について論じており、ジェスチャーは相手にある内容を伝えるという行為の達成を助けるために存在すると論じていた。その役割において具体的に、体の動きそのものが伝えるべき内容を表現する場合であること、体の動きによって会話の流れを調節することによって発話による情報のやりとりをスムーズにすると指摘しているため、こうした非言語的な情報を表現できる手法が必要がある。それらをふまえた上で本研究では、手や指の姿勢を検知することでこれを達成している。

また、Alexandra ら [13] によれば、こうした身体動作の同調が多く見られた対話はそうでな

い対話と比べてより親和的であったとしている。

### 5.3 ビデオ会議を改良した遠隔会議システム

福井ら [16] は、既存のビデオチャットにおいて誰が誰に注目して発話しているか認識できない問題に焦点を当て、カメラ画像からユーザーの顔の方向を判別し、その方向に存在する仮想空間中のアバターの表示を拡大することで誰に注目して発話しているかを認識可能にするシステムを開発した。本研究も同様の問題に対して、ユーザーとアバター間の頭部動作同期と、ボイスチャットで発話した際にアバターの口を動かすことによって誰に対して発話しているかわかりやすくする設計を行った。

Anthony ら [17] らはマイクアレイと Kinect を用いてビデオ会議を改良した遠隔会議に対して拡張現実を取り入れた研究を行った。複数のマイクから誰が発言しているのか認識をし、現在の話者以外をぼかして表示する、参加者の頭上に名前や発言時間を表示するなどの機能や、AR マーカーを利用してオブジェクトを表示したり組み合わせによって付加情報を表示する機能をもったシステムを開発した。本研究も同様にオブジェクトを表示することや、書き込みによる付加情報を提示するなど取り組んでいる。

### 5.4 仮想空間を用いた遠隔会議システム

本研究と同じく仮想空間内での遠隔会議システムを本題とした研究に Dean ら [18] らの研究がある。これはディスプレイの前に置かれた Kinect を用いて実世界のユーザーの動きをアバターに適用し、それをインターネットを介した仮想空間での会議で、頭部を実写画像にした棒人間のモデルに対して適用する。またユーザーはその仮想空間での会議をディスプレイで視聴する。本研究では、ディスプレイを用いることに伴う頭部の可動範囲の制約を解決している。この制約は、例えば仮想空間内で真横にいる人に対して話したい場合、ディスプレイを見ないと仮想空間の様子がわからないため必然的にディスプレイを見ることができる角度までしか頭を動かせないという制約がある。本研究ではヘッドマウントディスプレイを用い、装着したユーザがどこを向いていても映像が追従するため、この問題を解決する。

Lindeman ら [19] や Erickson ら [20] は SecondLife<sup>1</sup> を用いて遠隔会議を行った。SecondLife とは 3DCG で構成されたインターネット上に存在する仮想世界ユーザーはバーチャルな世界で好みのアバターになり、現実の世界とは異なる生活を送ることができるものである。Lindeman が行った会議では、SecondLife 内で 39 人が参加し約 9 時間ほど行われた。仮想空間上で仮想的なプロジェクタスクリーンを用いてプレゼンを行い、また参加者はアバターを操作して会議を行った。しかしこの会議では様々な問題が発生したと指摘している。まず、こうしたアバターを用いた会議ではアバターの操作するのが難しいという問題である。これにより会議において操作することにユーザが注力してしまい会議に集中できなくなってしまうという問題や、システムの未経験者には操作の難しさにより会議に用いようとするモチベーションが

---

<sup>1</sup>SecondLife, <http://secondlife.com/>

低くなってしまうという問題を引き起こす。また、プレゼンテーションにおいてボディランゲージやその他ジェスチャを用いる場合、システムで既に用意されている数種類のジェスチャを利用せざるを得ないため、表現の幅が狭い問題も発生した。本研究では、ユーザの実際の頭の動きや手や指の動きをそのままアバターの操作のための入力として用いているため、操作が難しいといったことは起こりにくい。また表現の幅に関してはユーザは自由に操作を行うことができるため表現の幅が狭い問題を解決している。

## 5.5 没入型仮想空間に関する研究

Young ら [21] は同一の物理的空間に存在する 2 人のユーザそれぞれに対して OculusRift を装着させ、自分を含めたお互いの動作を映像中でアバターで表現し実際にハイタッチさせその際の相互作用について研究した。フルボディの人型アバターを用いた場合と腕のみを用いて表現した場合に差があるか調査をしたものであるが、結果としてユーザの物理的な配置は重要ではないがアバターの形が重要であるという結論を得ていた。本研究でも同様に腕だけではなく仮想環境でフルボディの人型アバターを用いており、この事に対し効果があるということが言える。

Amores ら [22] は ShowMe というシステムを開発した。これは作業者と支援者の 2 人に分かれて使うシステムであり、両者はカメラが設置されたヘッドマウントディスプレイを装着する。映像を共有することで協調作業を行うものであり、作業者は自らの視界と手の動きを、支援者は手の動きのみをお互いの映像に共有することで作業を支援するものである。

Bodenheimer ら [23] は、ユーザに OculusRift を装着し、カメラから手足に動きを検知するための光を発する装置をとりつけ没入型仮想空間を提示しつつ自らの手足の動作をアバターの操作への入力とするシステムを開発した。これを用いてアバターを操作しブロックの端から下りることに対してアバターを用意した場合と用意しなかった場合、また棒人間のようなアバターと人型のアバターを用意しそれぞれにおいてどのような差が生じるかを研究した。結果としては自らが操作可能なアバターが存在することはブロックを下りるというタスクを完了するかどうかということに対してとても重要な情報であり、アバターがない場合に比べてタスクの完了時間に有意な差が現れた、一方でアバターの形状に関してはタスクの完了時間に有意な差はなかったとしている。またアバターが存在することは自らが仮想空間内でできることをユーザに示す役割も持つと結論づけており、本研究においてもアバターを用いることでユーザの操作可能な範囲を自らの操作するアバターの腕の届く範囲と暗に示すことによって操作に注力させてしまうリスクを減らしている。

## 第6章 結論

### 6.1 まとめ

本研究では参加者の情報を特定できない匿名の状態での会議に焦点を当てた遠隔会議システムを開発した。ユーザが没入型ヘッドマウントディスプレイを装着しハンドトラッキングシステムを仮想空間内のアバターの頭部と腕部を一人称視点から直感的に操作することを可能にする。ユーザの頭部の姿勢を検知し頷き否定などのジェスチャや自分や相手がどこに注視しているかをネットワークを介してお互いにわかるようになっている。また会議というドメインにおいて、ジェスチャが会話の流れを調節することによって発話による情報のやりとりをスムーズにする先行研究の知見に基づきハンドトラッキングによってアバターの手や指の姿勢を検知して柔軟な操作を可能としている。またボイスチャットでは通常のボイスチャットでは現実のユーザが特定されてしまう恐れがあることからリアルタイムに音声処理を施し匿名化することでこの問題に対応した。更に先行研究の知見からアバターの口を発話に合わせて動かすことにより誰が誰に向かって発話しているかを視認することを可能としている。会議として資料を用いて説明する場面を想定し資料の表示や操作、また相手との資料の共有を行う機能も実装している。またこの機能はキーボードやマウスなどを使わず、手の操作のみで資料を操作可能としている。

### 6.2 今後の展望と応用

本研究では、参加者を匿名した場合に焦点を当ててシステムを開発したが、機能としては資料提示やジェスチャなど参加者が特定できる場合でも有用な機能もある。例えばこれらの機能を用いた遠隔協調作業やe-learningなどの利用方法に対しても考察しそれをふまえて機能を拡張していく。また、リップシンク部分の実装の改善も今後解決すべき問題点の1つである。

研究を進めていく中で、遠隔会議をするときはGoogleDocument<sup>1</sup>などのオンラインで参加者と一緒に編集できるものを使いながら議論をすることや、テキスト入力によって議事録をとりながら議論をすることがあるという意見を得た。その意見を元に、新しい機能のプロトタイプとして仮想空間内にデスクトップを表示する機能を実装した。その機能を図6.1に示す。

本システムのヘッドマウントディスプレイを装着しているためキーボードが見えないという問題により、キーボードでタッチタイピングができる人にのみ有用な機能となってしまうが、実際に使用してみたところその条件を満たした人であれば十分実用的な機能になったと

---

<sup>1</sup>Google Document, <https://docs.google.com/>

考える。また Windows 上で動くすべてのアプリケーションを仮想空間内から操作できるため GoogleDocument に限らず、例えばペントアプレットを用いて絵を描く、ブラウジングをすることといったことが可能なほか、従来のアプリケーションの操作手法と変わらずそれらを行うこともメリットである

将来的な応用としてデスクトップ画面をネットワークを介して他参加者とデスクトップ画面を同期するなどの機能をつけ、アバターの操作機能と組み合わせた新しいインタラクションを模索していく。

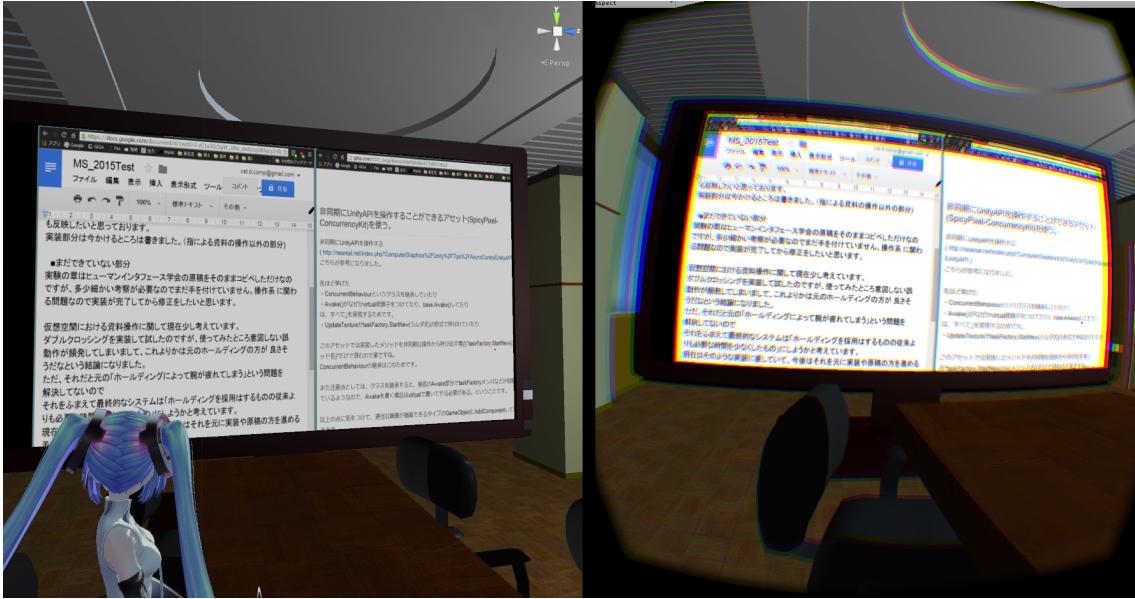


図 6.1: 左：仮想空間内にデスクトップを表示させている様子 右：左図においてユーザに見える映像

## 謝辞

本論文を執筆するにあたり、指導教員である田中二郎教授をはじめ、高橋伸准教授、志築文太郎准教授にはゼミを通して、丁寧なご指導とご助言を頂きました。深く感謝の意を表します。また、インタラクティブ・プログラミング研究室の皆様には、実験への協力や研究に対するアドバイスを頂き大変お世話になりました。この場を借りてご協力を頂いた関係者の皆様に感謝いたします。またシステム開発において心の支えであった初音ミクさんにも感謝いたします。最後に、6年間の大学生活を過ごすにあたって支えてくれた家族や、学生生活と共に過ごし様々な面でお世話になった友人にも感謝いたします。ありがとうございました。

## 参考文献

- [1] Carmen Egido. Video Conferencing As a Technology to Support Group Work: A Review of Its Failures. Proceedings of the 1988 ACM Conference on Computer-supported Cooperative Work (CSCW '88), pp. 13-24, 1988.
- [2] 玉木秀和, 東野豪, 小林稔, 井原雅行, 岡田謙一. 遠隔会議における発話衝突低減. 情報処理学会論文誌, pp. 1797-1806 , 2012.
- [3] Christian Vogler, Paula Tucker, Norman Williams. Mixed Local and Remote Participation in Teleconferences from a Deaf and Hard of Hearing Perspective. Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility, pp.30:1-30:5 , 2013.
- [4] 浅井俊晴, 中山裕美, 梶克彦, 河口信夫. SAMTK-3D を用いた多地点間仮想空間コミュニケーションとその応用. マルチメディア, 分散協調とモバイルシンポジウム 2011 論文集 , pp. 1664-1671 , 2011.
- [5] Aljosha Demeulemeester, Katriina Kilpi, Shirley Elprama, Sammy Lievens, Charles-Frederik Hollemeersch, An Jacobs, Peter Lambert, Rik Van de Walle. The ICOOCOON Virtual Meeting Room: A Virtual Environment As a Support Tool for Multipoint Teleconference Systems. Proceedings of the 11th International Conference on Entertainment Computing, pp.158-171, 2012.
- [6] Robert Sproull, Sara Kiesler. Reducing Social ContextCues: Electronic Mailin Organizational Communication. Management Science, pp.1492-1512, 1986.
- [7] Joseph Walther. Computer-mediated communication:Impersonal,interpersonal. and hyperpersonal interaction. Communication Research, pp. 3-43, 1996.
- [8] 中村卓, 高橋伸, 田中二郎. 大画面環境におけるハンドジェスチャの選択手法 : ダブルクロッシングの提案と他の選択手法との比較. 電子情報通信学会論文誌. D, 情報・システム, pp.978-988, 2013.
- [9] 四倉達夫, 武藤淳一, 今村達也, 藤井英史, 森島繁生. 仮想人物によるサイバースペース上で のコミュニケーションシステムの構築. 電子情報通信学会技術研究報告. HIP, ヒューマン情報処理 , pp.39-46, 1999.

- [10] 石井亮, 宮島俊光, 藤田欣也. アバタ音声チャットシステムにおける会話促進のための注視制御. ヒューマンインタフェース学会論文誌, pp.87-94 , 2008.
- [11] Roel Vertegaal and Yaping Ding. Explaining Effects of Eye Gaze on Mediated Group Conversations:: Amount or Synchronization?. Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work, pp.41-48, 2002.
- [12] 喜多壯太郎. ジェスチャーの認知科学 ひとはなぜジェスチャーをするのか. 認知科学, pp.9-21, 2000.
- [13] Paxton Alexandra and Dale Rick. Frame-differencing methods for measuring bodily synchrony in conversation. Behavior Research Methods, Vol. 45(2), pp. 329-343, 2013.
- [14] Yuki Inoue, Eisuke Ono, Jinhwan Kwon, Masanari Motohashi, Daisuke Ikari, Ken-ichiro Ogawa and Yoshihiro Miyake. Head motion synchronization in the process of consensus building. Proceedings of the 2013 IEEE/SICE International Symposium on System Integration, pp.70-75, 2013.
- [15] Koichi Takahashi, Hironobu Fukai and Yasue Mitsukura. "Head pose tracking with individual adjustment for communication tools", Proceedings of 4th International Conference on Human System Interaction, pp.386 -391, 2011.
- [16] 福井健太郎, 喜多野美鈴, 岡田謙一. 仮想空間を使った多地点遠隔会議システム: e-MulCS. 情報処理学会論文誌, pp. 3375-3384 , 2002.
- [17] Anthony DeVincenzi, Lining Yao, Hiroshi Ishii and Ramesh Raskar. Kinected Conference: Augmenting Video Imaging with Calibrated Depth and Audio. Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work , pp.621-624 , 2011.
- [18] Jesse Dean, Mark Apperley and Bill Rogers. Refining Personal and Social Presence in Virtual Meetings Proceedings of the Fifteenth Australasian User Interface Conference - Volume 150 , pp.67-75 , 2014.
- [19] Robert W. Lindeman, Reiners Dirk and Steed Anthony. Practicing what we preach: IEEE VR 2009 virtual program committee meeting. IEEE Computer Graphics and Applications, 29(2), pp. 80-83, 2009.
- [20] Thomas Erickson, N. Sadat Shami, Wendy A. Kellogg and David W. Levine. Synchronous Interaction Among Hundreds: An Evaluation of a Conference in an Avatar-based Virtual Environment. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, p.503-512, 2011.

- [21] Mary K. Young, John J. Rieser and Bobby Bodenheimer. Dyadic Interactions with Avatars in Immersive Virtual Environments: High Fiving. Proceedings of the ACM SIGGRAPH Symposium on Applied Perception, pp.119-126, 2015.
- [22] Judith Amores, Xavier Benavides and Pattie Maes. ShowMe: A Remote Collaboration System That Supports Immersive Gestural Communication. Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems, pp.1343-1348, 2015.
- [23] Bobby Bodenheimer and Qiang Fu. The Effect of Avatar Model in Stepping off a Ledge in an Immersive Virtual Environment. Proceedings of the ACM SIGGRAPH Symposium on Applied Perception, pp.115-118. 2015.

## 付録A 実験同意書及びアンケート用紙

次ページに、評価実験の際に用いた実験同意書及びアンケート用紙示す。

## **被験者実験承諾書**

**研究課題名 :**

仮想空間におけるメニュー操作の手法に関する実験

**実施責任者 :**

システム情報工学研究科コンピュータサイエンス専攻 2 年 根岸匠

**実験場所 :**

筑波大学総合研究 B 棟

**実験内容 :**

ヘッドマウントディスプレイ(以下 OculusRift)とハンドトラッキングシステム(以下 LeapMotion)を用いて  
仮想空間におけるメニューを操作する。

**実験課題の詳細 :**

3 種類の操作手法(※1)で仮想空間でのメニュー操作(※2)を行う。

**※1について**

- 1.)OculusRift をタップすることによってメニューを操作する。
- 2.)机の上に置かれた LeapMotion によってハンドトラッキングをし、それを用いてメニューを操作する。
- 3.)OculusRift に取り付けられた LeapMotion によってハンドトラッキングし、それを用いてメニューを操作する。

**※2について**

まず根岸から操作の説明を行う。

その後 Oculus と LeapMotion は貸出し、一度それぞれの手法を練習する。

それから実験を行う。実験の内容はドラッグ操作を行い、10 個の仮想空間のメニューを規定の位置へと操作するというものである。

**個人情報の扱いについて :**

収録したデータは本研究に直接関わる目的以外には使用しません。またいかなる場合にも被験者の個人情報が外部に出ることはあります。研究成果を論文としてまとめる場合には、被験者名は全て匿名(記号表記)とし、実験データを統計処理した数値とその記号表記のみを使用します。

以上の実験について、被験者として協力することを承諾します。また、個人情報の保護を条件に実験結果を論文などの形で公表することを承諾します。

日付 :

名前 :

所属 :

連絡先(電子メールまたは電話番号) :

## アンケート

### 問 1

OculusRift を使用するのは初めてですか？(適する方に○をつけてください) はい・いいえ

### 問 2

LeapMotion を使用するのは初めてですか？(適する方に○をつけてください) はい・いいえ

### 問 3

この中ならこれが一番「使い続けられそうだ」と思った手法はどれですか？(番号に丸をつけてください)

- 1 )OculusRift をタップすることによってメニューを操作する。
- 2 )机の上に置かれた LeapMotion によってハンドトラッキングをし、メニューを操作する。
- 3 )OculusRift に取り付けられた LeapMotion によってハンドトラッキングし、メニューを操作する。

### 問 4

この中ならこれが一番「操作がしやすい」と思った手法はどれですか？(番号に丸をつけてください)

- 1 )OculusRift をタップすることによってメニューを操作する。
- 2 )机の上に置かれた LeapMotion によってハンドトラッキングをし、メニューを操作する。
- 3 )OculusRift に取り付けられた LeapMotion によってハンドトラッキングし、メニューを操作する。

### 問 5

手法それぞれに関して操作しての感想があれば教えて下さい。些細なことでも結構です。

(トラッキングが外れやすい、こうできたらもっとよいなど…)  
※欄が足りない場合は裏へお願いします。

1)OculusRift をタップすることによってメニューを操作する。

2)机の上に置かれた LeapMotion によってハンドトラッキングをし、メニューを操作する。

3)OculusRift に取り付けられた LeapMotion によってハンドトラッキングし、メニューを操作する。

実験に協力していただきありがとうございました。

実験者：根岸 匠