

# 公立はこだて未来大学 2023 年度 システム情報科学実習 グループ報告書

Future University Hakodate 2023 Systems Information Science Practice  
Group Report

## プロジェクト名

気持ちと世界をつなぐ X-Reality とスマートウォッチ

## Project Name

X-Reality and Smartwatch Connecting Your Feelings and the World

## グループ名

前期：映像班/スマートウォッチ班/解析班

後期：A 班/B 班/C 班

## Group Name

first semester: Video Group/Smartwatch Group/Analysis Group

second semester: A Group/B Group/C group

## プロジェクト番号/Project No.

1

## プロジェクトリーダー/Project Leader

前期：新田蒼心

後期：メンバ全員, 交代制

first semester: Soshi Nitta

second semester: All of member, Shift system

## グループリーダー/Group Leader

前期：品川誠剛, 丹下裕太

後期：新田蒼心, 品川誠剛, 川端勇雅

first semester: Seigo Shinagawa, Yuta Tange

second semester: Soshi Nitta, Seigo Shinagawa, Yuga Kawabata

## グループメンバ/Group Member

メンバリスト\*参照

## 指導教員

佐藤仁樹教授 新美礼彦教授

## Advisor

Prof. Hideki Satoh Prof. Ayahiko Niimi

## 提出日

2024 年 1 月 22 日

## Date of Submission

January,22th,2024

## メンバリスト\*

川端勇雅	Yuga Kawabata
品川誠剛	Seigo Shinagawa
高杉拓海	Takumi Takasugi
丹下裕太	Yuta Tange
長澤颯音	Hayato Nagasawa
新田蒼心	Soshi Nitta
松本柊哉	Shuya Matsumoto
結城熙乙	Hiroto Yuki
米森麻翔	Manato Yonemori

## 概要

本プロジェクトは VirtualReality(以下, VR と表記する)とスマートウォッチ(以下, SW と表記する)を利用して「自分の気持ちと世界をつなぐ」ことを目標とした。A 班:VR で街並みを表示しウォーキングマシンの上を歩くことで疑似的な観光を再現し, その際に得られた運動データを解析することでユーザの運動レベルを計り楽しく運動をするシステム。B 班:VR に高所や深海などの恐怖症に対応した映像を表示し, それを見た人の心拍数を計測, 解析して恐怖の対象を観測し恐怖症であるか否かを判断するシステム。C 班:安眠を促進する映像や音楽を表示した VR を装着 (VR を外してプラネタリウムでも可)して睡眠を行い, 睡眠時の心拍数を計測, 解析しノンレム睡眠時にアラームを鳴らすことによりユーザ自身の睡眠活動を改善するシステム。これらのシステムにより, ユーザの身体データを解析することで情動の推定, 生活の質の向上を実現できる可能性があることがわかった。しかし, 解析に必要な量のデータが不足しているため, 結果の信憑性には欠ける点を留意しなければならない。

**キーワード** VR, スマートウォッチ, ANT+, 主成分分析, 重回帰分析, パターン認識, 運動, 恐怖症, 睡眠

(※文責: 長澤颯音)

# Abstract

This project aims to connect own emotions and the world using Virtual Reality (VR) and Smart Watches (SW). Group A: A system that displays a cityscape in VR, reproduces sightseeing on a walking machine, and analyzes the exercise level. Group B: A system that displays images corresponding to phobias such as fear of heights and the deep sea in VR, and analyzes the heart rate of the viewer whether or not the viewer has the phobia. Group C: A system to improve the user's sleep activity using the heart rate during sleep and an alarm clock. These systems have shown the possibility of estimating emotions and improving the quality of life by analyzing the user's physical data. However, it should be noted that the results lack reliability due to the lack of the necessary amount of data for analysis.

**Keyword** VR, smartwatch, ANT+, principal component analysis, Multiple regression analysis, pattern recognition, exercise, phobia, sleep

(※文責: 新田蒼心)

# 目次

<b>第 I 部</b>	<b>前期</b>	<b>1</b>
<b>第 1 章</b>	<b>はじめに</b>	<b>2</b>
1.1	前年度の成果 . . . . .	2
1.2	グループ概要 . . . . .	2
1.2.1	映像班 . . . . .	2
1.2.2	スマートウォッチ班 . . . . .	2
1.2.3	解析班 . . . . .	3
1.3	活動計画 . . . . .	3
<b>第 2 章</b>	<b>映像班</b>	<b>4</b>
2.1	技術習得の概要 . . . . .	4
2.1.1	到達目標 . . . . .	4
2.1.2	課題設定と解決に向かって . . . . .	4
2.2	実習内容 . . . . .	6
2.2.1	VR のステージ制作 . . . . .	6
2.2.2	ゲームの仕様 . . . . .	6
<b>第 3 章</b>	<b>スマートウォッチ班</b>	<b>8</b>
3.1	前年度の成果 . . . . .	8
3.2	目的達成に向けて . . . . .	8
3.2.1	技術の学習・習得 . . . . .	8
3.3	アプリケーション概要 . . . . .	8
3.3.1	使用したプログラムと機材 . . . . .	8
3.3.2	ソフトウェアの仕様 . . . . .	9
3.4	実習内容 . . . . .	9
3.4.1	リアルタイムで心拍数を表示 . . . . .	9
3.4.2	心拍データの書き出し . . . . .	10
<b>第 4 章</b>	<b>解析班</b>	<b>12</b>
4.1	技術習得の概要と課題 . . . . .	12
4.2	課題解決に向けて . . . . .	12
4.2.1	テーマに沿った解析手法の検討 . . . . .	12
4.2.2	感情診断用のプログラムの作成 . . . . .	12
4.3	3 つの統計学的手法 . . . . .	13
4.3.1	主成分分析 . . . . .	13
4.3.2	重回帰分析 . . . . .	14
4.3.3	パターン認識 . . . . .	16

<b>第 5 章</b>	<b>爆弾ゲーム</b>	<b>20</b>
5.1	作成の経緯 . . . . .	20
5.2	爆弾ゲーム・感情診断の流れ . . . . .	20
5.3	感情診断について . . . . .	21
<b>第 6 章</b>	<b>中間発表会</b>	<b>25</b>
6.1	発表の流れ . . . . .	25
6.2	発表評価 . . . . .	25
6.2.1	評価シートについて . . . . .	25
6.2.2	評価内容について . . . . .	26
6.2.3	フィードバックを受けて . . . . .	26
6.2.4	中間発表会後の自己評価 . . . . .	27
<b>第 7 章</b>	<b>まとめ</b>	<b>28</b>
7.1	各グループのまとめ . . . . .	28
7.1.1	映像班のまとめ . . . . .	28
7.1.2	スマートウォッチ班のまとめ . . . . .	29
7.1.3	解析班のまとめ . . . . .	29
7.2	反省と今後の展望 . . . . .	29
7.2.1	各メンバの反省 . . . . .	29
7.2.2	今後の展望 . . . . .	30
<b>第 II 部</b>	<b>後期</b>	<b>32</b>
<b>第 8 章</b>	<b>はじめに</b>	<b>33</b>
8.1	メンバの編成 . . . . .	33
8.2	プロジェクトリーダー . . . . .	33
8.3	グループ概要 . . . . .	34
8.3.1	A 班 . . . . .	34
8.3.2	B 班 . . . . .	34
8.3.3	C 班 . . . . .	34
8.4	活動計画 . . . . .	34
8.4.1	A 班 . . . . .	35
8.4.2	B 班 . . . . .	35
8.4.3	C 班 . . . . .	36
<b>第 9 章</b>	<b>A 班</b>	<b>38</b>
9.1	テーマの決定 . . . . .	38
9.2	成果物紹介 . . . . .	38
9.2.1	映像技術 . . . . .	39
9.2.2	プレイの手順 . . . . .	39
9.2.3	解析の流れ . . . . .	40
9.2.4	解析技術 . . . . .	40

9.3	使用した技術 . . . . .	41
9.4	仕様書 . . . . .	41
9.4.1	ソフトウェア名 . . . . .	41
9.4.2	開発環境 . . . . .	41
9.4.3	動作環境 . . . . .	41
9.4.4	ディレクトリ構成とソースコード . . . . .	42
9.4.5	各ステージの詳細な解説 . . . . .	45
9.4.6	解析 . . . . .	46
9.5	まとめ . . . . .	46
9.5.1	映像担当のまとめ . . . . .	47
9.5.2	スマートウォッチ担当のまとめ . . . . .	47
9.5.3	解析担当のまとめ . . . . .	47
9.5.4	各メンバの反省 . . . . .	47
<b>第 10 章</b>	<b>B 班</b>	<b>49</b>
10.1	目的 . . . . .	49
10.2	テーマの決定 . . . . .	49
10.3	成果物 . . . . .	49
10.3.1	心拍数受信・書き出しシステム . . . . .	50
10.3.2	アプリケーション . . . . .	50
10.3.3	解析プログラム . . . . .	52
10.4	使用した技術 . . . . .	52
10.4.1	映像技術 . . . . .	52
10.4.2	3DCG 技術 . . . . .	52
10.4.3	解析技術 . . . . .	53
10.5	アプリケーション仕様書 . . . . .	57
10.5.1	概要 . . . . .	57
10.5.2	ディレクトリ構造 . . . . .	58
10.5.3	プログラム詳細 . . . . .	59
10.6	解析プログラム仕様書 . . . . .	60
10.6.1	概要 . . . . .	60
10.6.2	解析プログラムの開発環境 . . . . .	61
10.6.3	プログラムの詳細 . . . . .	61
10.7	まとめ . . . . .	62
10.7.1	映像担当のまとめ . . . . .	63
10.7.2	スマートウォッチ担当のまとめ . . . . .	63
10.7.3	解析担当のまとめ . . . . .	63
10.7.4	各メンバの反省 . . . . .	63
<b>第 11 章</b>	<b>C 班</b>	<b>64</b>
11.1	目的・目標 . . . . .	64
11.2	テーマの決定 . . . . .	64
11.3	成果物 . . . . .	65

11.3.1	アプリケーション	65
11.3.2	成果物の全体の流れ	65
11.4	使用した技術	67
11.4.1	Unity で使用した機能	67
11.4.2	DateTime 構造体	67
11.4.3	PlayerPrefs	68
11.4.4	ドロップダウン	68
11.4.5	解析技術	69
11.5	仕様書	70
11.5.1	概要	70
11.5.2	ディレクトリ構成	71
11.5.3	ソースコード詳細	71
11.5.4	解析技術の仕様書	74
11.6	まとめ	75
11.6.1	映像担当のまとめ	76
11.6.2	スマートウォッチ担当のまとめ	76
11.6.3	解析担当のまとめ	76
11.6.4	各メンバの反省	76
<b>第 12 章</b>	<b>成果発表会</b>	<b>77</b>
12.1	成果発表会に向けて	77
12.2	ポスター・スライドの作成	77
12.3	成果発表会の発表形式	77
12.4	フィードバック	77
12.4.1	発表技術のフィードバック	78
12.4.2	プロジェクトのフィードバック	78
12.4.3	フィードバックを受けて	79
12.4.4	成果発表会後の自己評価	79
<b>第 13 章</b>	<b>まとめ</b>	<b>81</b>
13.1	各グループのまとめ	81
13.1.1	A 班のまとめ	81
13.1.2	B 班のまとめ	81
13.1.3	C 班のまとめ	82
13.2	プロジェクトのまとめ	83
13.2.1	プロジェクトのまとめ	83
13.3	反省と今後の展望	83
13.3.1	各メンバの反省	83
<b>参考文献</b>		<b>85</b>
<b>付録 A</b>	<b>使用した技術について</b>	<b>86</b>
A.1	Unity	86

	A.1.1	概要	86
	A.1.2	成果物への関与	86
A.2		ANT+	87
	A.2.1	概要	87
	A.2.2	成果物への関与	87
A.3		Node.js	87
	A.3.1	概要	87
	A.3.2	成果物への関与	88
A.4		Python	88
	A.4.1	概要	88
	A.4.2	成果物への関与	88

第 I 部

前期

# 第 1 章 はじめに

この章では、前期における本プロジェクトの目的やグループごとの概要について記述する。

(※文責: 新田蒼心)

## 1.1 前年度の成果

本プロジェクトは、「X-Reality とスマートウォッチを用いて身の回りの現象に接続し、自分の気持ちと世界をつなぐための情報リンクを作成する」ことを目的とするプロジェクトである。その目的を達成するには、様々な技術を習得する必要があった。そのため、前期では目的の達成につながる技術の習得を目標とした。具体的には、技術の分野ごとにグループ分けを行い、それぞれに対応した技術を習得、最後に各グループの技術を統合した試作品の作成を行うことである。

(※文責: 長澤颯音)

## 1.2 グループ概要

この章では、各グループの概要と目標について記述する。

(※文責: 長澤颯音)

### 1.2.1 映像班

本プロジェクトにおける、「X-Reality」に対応する技術を習得するグループである。主に、「Virtual Reality」(以後、VR と記載) や「Augmented Reality」についての技術を習得を目標とした。そのために、Unity を用いた VR アプリの制作、360 度カメラの撮影、映像の取り込み方、フォトメトグラリ (2. 1. 2 参照) を用いた 3D モデルの生成などを行った。

(※文責: 長澤颯音)

### 1.2.2 スマートウォッチ班

本プロジェクトにおける、「スマートウォッチ」に対応する技術を習得するグループである。主に、スマートウォッチから得られた身体データを PC にリアルタイムで送信する技術、身体データを PC に保存する技術の習得を目標とした。

(※文責: 長澤颯音)

### 1.2.3 解析班

本プロジェクトにおける、「情報リンクの作成」に対応する技術を取得するグループである。主に、スマートウォッチ班から送られた身体データを認識・分類する技術の習得を目標とした。特に、CSV ファイルに変換されたデータを重回帰分析やパターン認識、主成分分析を行って解析した。

(※文責: 長澤颯音)

## 1.3 活動計画

プロジェクト開始から 1 ヶ月間を技術習得月間とし、各班ごとに必要な技術を習得を行った。詳しい技術の習得内容については、以降それぞれの班によって記述する。また、その後の活動計画については以下にまとめる。

- 活動計画
  - ・各班で取得した技術を共有し、それぞれの技術を用いて作ることでできる試作品を立案する。
  - ・試作品の製作
  - ・立案した試作品を元に試作品を実際に制作していく。
  - ・中間発表会、中間報告書の作成
  - ・今回行った作業、技術をまとめたスライドや、報告書を作成する。
  - ・次回の課題や改善点の特定

(※文責: 長澤颯音)

## 第 2 章 映像班

この章では、前期における映像班の活動内容を記述する。

(※文責: 新田蒼心)

### 2.1 技術習得の概要

この節では、映像班が設定した目標と習得した技術の概要について記述する。

(※文責: 新田蒼心)

#### 2.1.1 到達目標

映像班では、制作物における映像面でのシステムの構築、クオリティ向上を目標とした。したがって、その上で必要となるであろう技術を事前に洗い出し、学習を行った。

(※文責: 品川誠剛)

#### 2.1.2 課題設定と解決に向かって

##### Unity の環境構築&基礎の定着

各自で Unity Hub のインストールを行い、Unity Editor を起動して動作することを確認。その後基礎の学習のため、「Unity ちゃんが教える！初心者向け Unity 講座」[1]を見て学習し、Unity Editor の使い方、オブジェクトの操作、マテリアルとテクスチャについてなど、幅広く学んだ。

(※文責: 品川誠剛)

##### 360 度映像・3D キャプチャの理解

- 360 度映像
  - － 360 度カメラで撮った映像を、球体オブジェクトの内側のテクスチャとして取りこむことで、Unity で 360 度映像を実現できるようにした。
- フォトグラメトリ
  - － フォトグラメトリとは、複数枚の画像から、専用のソフトウェアを用いて 3D モデルを生成する手法のことである。
  - － まず、RICOH THETA S2 で 360 度動画を撮影、ffmpeg によって連番の画像に変換した。そして、変換された画像を Reality Capture に取り込み、3D モデルを生成した [3]。

(※文責: 品川誠剛)

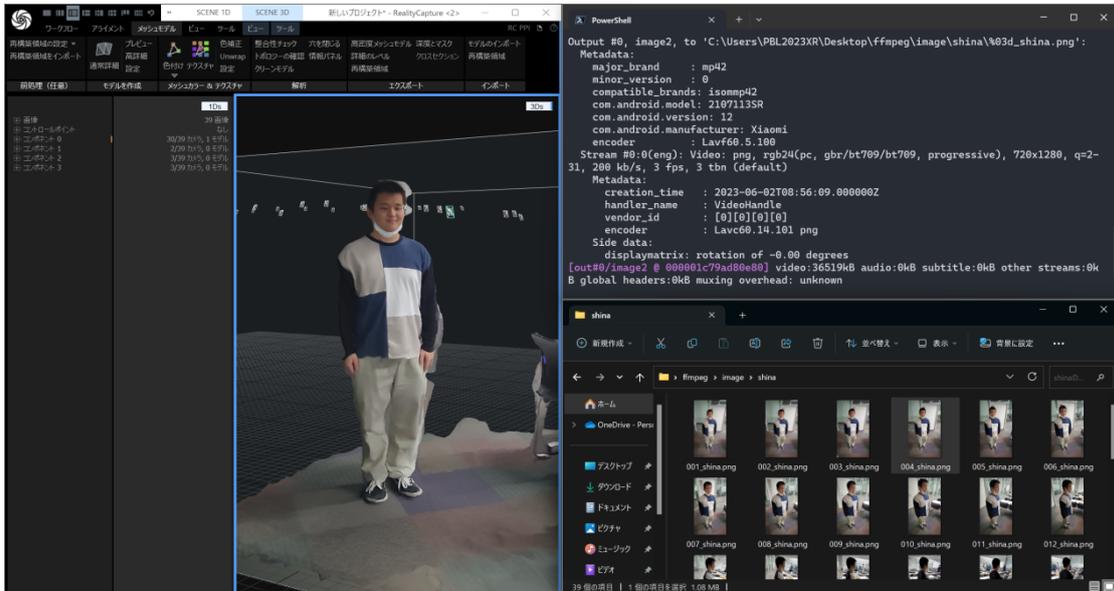


図 2.1 ffmpeg による連番画像の生成と Reality Capture における 3D モデル生成

## Unity と VR の連動

Unity Editor で作ったプロジェクトを Oculus Quest 2 で動かすことを目標として次のことを行った。

1. Oculus Link を PC にインストール, Oculus アカウントを作成。
2. VR の機種の一つである Oculus Quest 2 を PC と接続。
3. 専用アプリと連動して開発者モードを有効化する [2]。

これらの手順を行い, Oculus Quest 2 と PC をケーブルで繋ぐことで, Unity で作ったモノを VR 上で動かすことを可能にした。

(※文責: 品川誠剛)

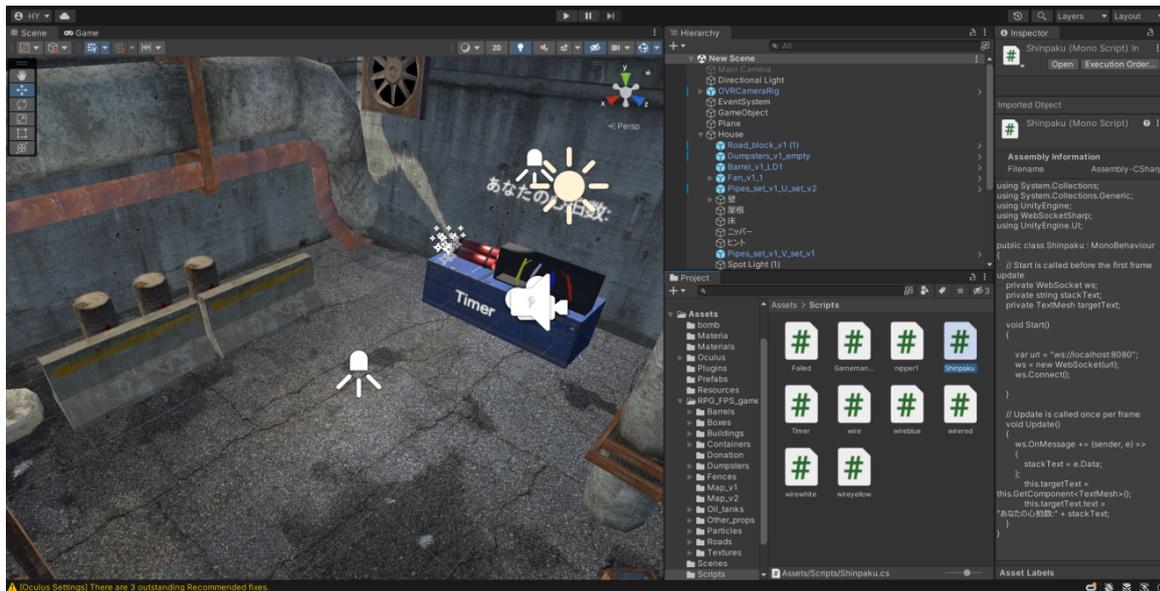


図 2.2 Unity Editor でのステージ制作

## 2.2 実習内容

学習期間で触れた技術の中で、VR を用いて 1 つの成果物を制作することに決定された。その後、プロジェクト全体で協議した結果、爆弾解除を VR 空間の中で行うゲームを作ることに決定した（5 章にて後述する）。この節では、制作した成果物（爆弾ゲーム）における、映像班が担当した箇所について記述する。

（※文責：結城熙乙）

### 2.2.1 VR のステージ制作

- Unity における GameObject を組み合わせて部屋を作り、Asset Store より入手したフリーのテクスチャを貼り付けた。
- 自前で 3D モデリングした爆弾を部屋の隅に設置し、横に解除用のワイヤーを設置した。
- ペイントソフトで作成した爆弾解除のためのヒントを Unity に取り込んで、爆弾のある反対側の壁に貼り付けた。
- その他、雰囲気を作るために、Asset Store より入手したアセット類（換気口、パイプラインなど）を部屋の中に配置した。

実際の成果物に向けて上記の作業を分担し、Unity Editor を用いてゲームのステージを作成した。

（※文責：結城熙乙）

### 2.2.2 ゲームの仕様

- ゲームは VR ゴーグル (Oculus Quest 2) およびスマートウォッチ (GARMIN vivosmart4) を装着してプレイする。

## X-Reality and Smartwatch Connecting Your Feelings and the World

- ゲームは Unity Editor 及び node.js がインストールされた環境の PC で実行される必要がある。
- node.js にて記述された，スマートウォッチで計測した心拍数を取得するためのプログラムを，バックグラウンドで実行する。
- プレイ中はプレイヤーの心拍数，およびゲーム開始からの経過時間が表示される。
- プレイヤーはコントローラを握り，ヒントを参考にしつつ，爆弾の解除ワイヤーを順番通りに切除する．正しく順番通りに切除することができればクリアとなる。
- クリア時に経過時間のタイマーが止まり，ゲームのクリアタイムが分かるようになっている。

(※文責: 結城熙乙)

## 第3章 スマートウォッチ班

この章では、スマートウォッチ班が行った活動内容を記述する。

(※文責: 松本柊哉)

### 3.1 前年度の成果

本グループでは、解析班と映像班の制作物で使用するための身体データを取得することを目的とした。また、取得するデータは映像班の制作したゲームのプレイ画面で表示を行うため、リアルタイムである必要があった。そのため、取得する身体データは、ANT+ 規格でリアルタイム送信ができる心拍数を対象とした。

(※文責: 松本柊哉)

### 3.2 目的達成に向けて

この節では、3.1 で設定した目的を達成するためのプロセスの詳細を述べる。

(※文責: 松本柊哉)

#### 3.2.1 技術の学習・習得

本プロジェクトでは、5月を技術習得月間とし、勉強会を行った。本グループでは、スマートウォッチで計測できる身体データをリアルタイムで取得すること目的とした。昨年プロジェクトの資料と参考サイト [4] を参考にすることで、Node.js を利用し、リアルタイムにデータ取得することができ、目的を達成した。

(※文責: 松本柊哉)

### 3.3 アプリケーション概要

この節では身体データの取得に関する概要について述べる。

(※文責: 松本柊哉)

#### 3.3.1 使用したプログラムと機材

ANT+ 経由で心拍数を送信し、可視化するために必要なものを以下に記す。

- 使用したプログラム

- ・ zadig v2.0.1.160.7z (ANT+ 用の Dongle を Windows で使用する為のドライバー)
  - ・ node-v18.16.0-x64.msi (Node.js のインストーラー)
  - ・ ant-plus-master(ANT+ 規格のデバイスと通信するためのソフトウェア)
  - ・ HeartBeats-master(心拍数を受信する為のソフトウェア)
  - ・ StartHeartBeats.bat(上記のソフトウェア内にある受信できているか確認できるプログラム)
  - ・ html/index.html(同じく上記のソフトウェア内にある html でリアルタイムで可視化させたアドレスバー)
- 使用した機材
    - ・ vivosmart 4
    - ・ ANT+ に対応した Dongle
    - ・ Windows 10/11

(※文責: 松本柊哉)

### 3.3.2 ソフトウェアの仕様

ANT+ に対応したスマートウォッチと USB Dongle を用意する。USB Dongle を PC に差し込み、ソフトウェアを立ち上げると自動的にスマートウォッチに接続され、Node.js を通してリアルタイムでグラフに起こした物が html ファイルで表示される。

(※文責: 松本柊哉)

## 3.4 実習内容

この節では、スマートウォッチ班が習得した技術の実習について記述する。実習内容は、昨年プロジェクト学習期末報告書 [5] を参考にした。今年度の作業に合わせ、node.js のバージョンの記述を更新し、それ以外を引用した。以下は、昨年の報告書を元に行なった実習内容である。

(※文責: 新田蒼心)

### 3.4.1 リアルタイムで心拍数を表示

#### セットアップ

1. C++ 環境インストールのため、Visual Studio のホームページ [15] にアクセスする。
2. 「Visual Studio のダウンロード」をクリックし、「Community 2022」を選択。
3. 自動的に「VisualStudioSetup.exe」がダウンロードされるので、そちらを開き「続行」をクリック。
4. インストールをクリック。
5. Python の公式ページ [16] にアクセスする。
6. 「Downloads」から Download for Windows の下にある「Python 3.11.1」をクリック。

7. 「python-3.11.1-amd64.exe」を開き、「Install now」をクリックする。
8. ANT+ 用の Dongle を Windows で使用するために Zadig というドライバーをインストール。（ここで zadig v2.0.1.160.7z をインストール）
9. 拡張子が 7z のため解凍ソフトなどを利用して解凍する。
10. Dongle を繋ぎ、Zadig を起動させる。
11. ドライバー上記の Options をクリックしその中にある List All Devices にチェックを入れる。
12. 上の枠を ANT USBStick2 を選択。
13. Replace Driver を選択して ANT+ 用のドライバーをインストール。
14. これでドライバーのインストールは終了だが、作業を Node.js で行うため Node.js をインストールする。
15. Node.js の HP よりインストーラーをダウンロード。
16. ダウンロードしたインストーラーを展開し、インストールを行う。
17. Next をクリック、承認のチェックを入れ、Next をクリック、インストール先を選定し、Next をクリック、インストールしたいコンポーネントを選択し、Next をクリック、インストールボタンをクリックし、インストールを開始する。
18. Node.js のインストールは完了したが、確認のためコマンドプロンプトを起動。
19. `node -version` のコマンドより、Node.js が入っていることとバージョンを確認。現時点では、v18.16.0 で動作確認が取れている。

### ソフトの起動

1. GitHub から tokjin/HeartBeats をダウンロード（zip ファイルのため解凍も忘れず）
2. コマンドプロンプト or PowerShell を開く
3. `cd ***` で、先ほど GitHub からダウンロードしてきたフォルダに移動。
4. `npm install` を実行。
5. 心拍計の電源を入れる。
6. 同封の StartHeartBeats.bat を開く（もしくは `node main.js` を実行）
7. うまく動いていれば count の数値が常に増えていく。
8. 先ほど GitHub からダウンロードしてきたフォルダの `html/index.html` を開くと以下のように表示される。

### ソフト内の説明

- count：心拍計のデータを読み取った数
- beat：実際に計測された心拍数
- id：心拍計の端末に割り当てられた固有 ID
- timeout：count が 10 秒間増えなかった場合タイムアウトしソフトが再起動する。

### 3.4.2 心拍データの書き出し

1. 昨年プロジェクトのファイルより、heartbeat-js.zip をダウンロードし、ドライバをインストールする。

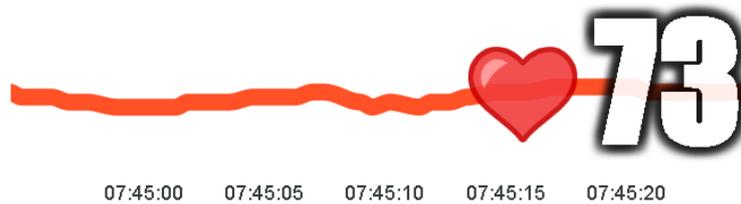


図 3.1 心拍数の表示

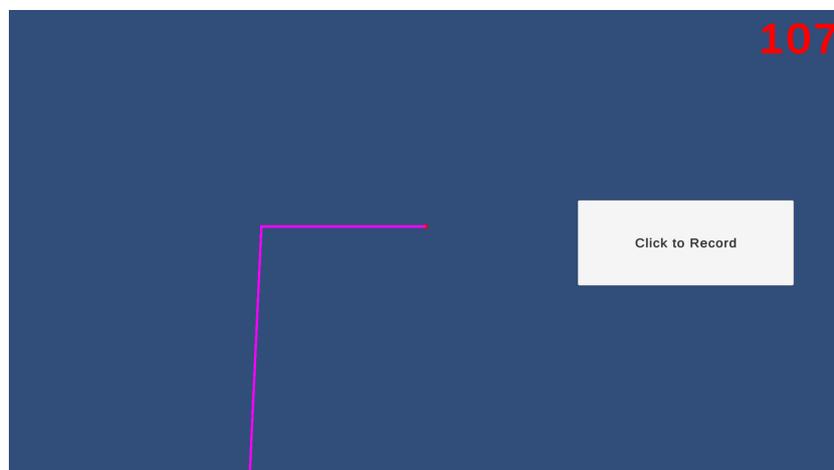


図 3.2 心拍数の記録

2. コマンドプロンプト or PowerShell を開く
  3. cd \*\*\*で、ダウンロードしたフォルダに移動.
  4. npm install ant-plus を実行.
  5. コマンドプロンプト or PowerShell を閉じる。
  6. 心拍計の電源を入れる.
  7. フォルダ内の mainjs.bat を起動する.
  8. 昨年プロジェクトのファイルより、exp\_build\_01 フォルダをダウンロードし、その中の ANT-plus.exe を起動した.
  9. 起動すると、以下のように表示される.
- 
10. Click to Record を押すと、心拍数が csv ファイルに記録される.

(※文責: 川端勇雅)

## 第 4 章 解析班

この章では、解析班の前期の活動内容について述べる。

(※文責: 米森麻翔)

### 4.1 技術習得の概要と課題

本グループは、スマートウォッチ班が計測した身体データを統計学的手法を用いて解析するグループである。目的達成のために主成分分析、重回帰分析、マハラノビス距離を用いたパターン認識の3つの手法について習得する必要がある。その上で、SW 班が計測したデータを解析し PC 上で可視化することを課題とした。

(※文責: 米森麻翔)

### 4.2 課題解決に向けて

この節では、4.1 で設定した課題の解決プロセスの詳細を述べる。

(※文責: 米森麻翔)

#### 4.2.1 テーマに沿った解析手法の検討

プロジェクトメンバで話し合いの末決定した「爆弾ゲーム（5 章にて後述する）を行った結果得られた心拍データをもとに感情診断を行う」というテーマに沿った解析手法を検討した。その結果、スマートウォッチ班が計測したデータに対し主成分分析を行いデータの次元を圧縮し、そのデータにパターン認識を行うことで診断結果を導出できると考えた。

(※文責: 米森麻翔)

#### 4.2.2 感情診断用のプログラムの作成

感情診断を行うため、主成分分析とマハラノビス距離によるパターン認識を行うプログラムを Python を用いて作成した。スマートウォッチから得られる心拍数データは 30 秒間で約 120 次元と膨大になるため、主成分分析を行うプログラムを用いて 4 次元にまで圧縮し、圧縮結果を csv ファイルに出力させた。次にパターン認識を行うプログラムを用いて、前述のファイルを分析し、その結果を csv ファイルに出力させた。

(※文責: 高杉拓海)

### 4.3 3つの統計学的手法

この節では、5月の勉強会で学んだ3つの統計学的手法の詳細について述べる。

(※文責: 米森麻翔)

#### 4.3.1 主成分分析

佐藤(2023)によると、主成分分析とはデータに内在する情報という概念を分散で捉えて、情報をできるだけ失うことなく、より低次元の新たな変数を定義していく手法である[6]。以下に主成分分析のアルゴリズムを示す。

$$x_{L_n} : n \text{ 番目の学習データ}, \quad x_L = (x_{L_1}, \dots, x_{L_{d_x}})^T \quad (4.1)$$

$$x_{E_n} : n \text{ 番目の評価データ}, \quad x_E = (x_{E_1}, \dots, x_{E_{d_x}})^T \quad (4.2)$$

$$d_x : \text{説明変数の次元} \quad (4.3)$$

(1)  $S_L$  を求める

$$S_L : x_L \text{ の共分散行列} \quad (4.4)$$

$$S_L = \begin{pmatrix} S_{11} & \cdots & S_{1d_x} \\ \vdots & \ddots & \vdots \\ S_{d_x 1} & \cdots & S_{d_x d_x} \end{pmatrix} \quad (4.5)$$

(2)  $\lambda_S, M$  を求める

$$\lambda_S : S_L \text{ の固有値}, \quad |\lambda_{S[i]}| \geq |\lambda_{S[i+1]}|, \quad 1 \geq i \geq d_x \quad (4.6)$$

$$M : \text{主軸行列}, \quad M \equiv [e_{S_1}, \dots, e_{S_{d_x}}] \quad (4.7)$$

$$e_{S_i} : S_L \text{ の } i \text{ 番目の固有ベクトル} \quad (4.8)$$

(3)  $y_L, y_E$  を求める

$$y_L : x_L \text{ の主成分ベクトル}, \quad y_L = M^T x_L \quad (4.9)$$

$$y_E : x_E \text{ の主成分ベクトル}, \quad y_E = M^T x_E \quad (4.10)$$

以上が主成分分析のアルゴリズムである。

(※文責: 米森麻翔)

次に、作成したプログラムの動作確認を行う。

デバッグ用の実行結果 [\*]

/\*----- Eigen Value for Learn data -----\*/

1.254249e+01 7.018927e+00 1.993934e+00 5.889831e-01 3.744076e-01

4.676227e-02 1.477315e-02 7.675563e-03 4.043429e-03 3.104814e-03

1.614123e-03 8.759492e-04 7.266429e-04 5.851757e-04 2.016857e-04

1.224975e-04 4.303526e-05 1.118877e-05 7.095131e-06 2.395594e-06

作成したプログラムの実行結果

/\*----- Eigen Value for Learn data -----\*/

1.254249e+01 7.018927e+00 1.993934e+00 5.889831e-01 3.744076e-01

4.676227e-02 1.477315e-02 7.675563e-03 4.043429e-03 3.104814e-03

1.614123e-03 8.759492e-04 7.266429e-04 5.851757e-04 2.016857e-04

1.224975e-04 4.303526e-05 1.118877e-05 7.095131e-06 2.395594e-06

上記の結果において、主成分ベクトルの値が一致していることがわかる。

以上が作成したプログラムの動作確認である。

(※文責: 丹下裕太)

### 4.3.2 重回帰分析

佐藤（2023）によると、未知のシステムを以下のような線形回帰モデルで近似するとき、その回帰係数を推定することを、重回帰分析と呼ぶ [7].

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p + \epsilon \quad (4.11)$$

$$\beta_0, \dots, \beta_p : \text{回帰係数} \quad (4.12)$$

$$\epsilon : \text{近似誤差} \quad (4.13)$$

以下に重回帰分析のアルゴリズムを示す。

(1) 学習データの係数行列  $B$  と、切片  $b_0$  を求める。

$$S_{xx} : x_L \text{ と } x_L \text{ の共分散行列} \quad (4.14)$$

$$S_{yx} = \begin{pmatrix} S_{11} & \cdots & S_{1d_x} \\ \vdots & \ddots & \vdots \\ S_{d_y 1} & \cdots & S_{d_y d_x} \end{pmatrix} \quad (4.15)$$

$$B = S_{yx} S_{xx}^{-1} \quad (4.16)$$

$$b_0 = \bar{y} - B\bar{x} \quad (4.17)$$

(2) 重回帰式を求める。

$$\tilde{y}_{L(n)} = BxL(n) + b_0 \quad | \quad n = 1, \dots, N \quad (4.18)$$

$$\tilde{y}_{E(n)} = BxE(n) + b_0 \quad | \quad n = 1, \dots, N \quad (4.19)$$

(3) 絶対値誤差を求める。

$$|\epsilon_L| = |\tilde{y}_L - y_L| \quad (4.20)$$

$$|\epsilon_E| = |\tilde{y}_E - y_E| \quad (4.21)$$

(4) 目的変数の平方誤差を求める.

$$\varepsilon_{i;L} = \left( \left( \sum_{n=1}^N (y_{i;L}[n] - \tilde{y}_{i;L}[n])^2 \right) / N \right)^{1/2}, \quad for \forall i \quad (4.22)$$

$$\varepsilon_{i;E} = \left( \left( \sum_{n=1}^N (y_{i;E}[n] - \tilde{y}_{i;E}[n])^2 \right) / N \right)^{1/2}, \quad for \forall i \quad (4.23)$$

(5)  $\varepsilon_L, \varepsilon_E$  を求める.

$$\varepsilon_L \equiv ((\varepsilon_{1;L}^2 + \dots + \varepsilon_{1;L}^y) / d_y)^{1/2} \quad (4.24)$$

$$\varepsilon_E \equiv ((\varepsilon_{1;E}^2 + \dots + \varepsilon_{1;E}^y) / d_y)^{1/2} \quad (4.25)$$

以上が、重回帰分析のアルゴリズムを数式で記述したものである.

次に、作成したプログラムの動作確認を行う.

デバッグ用の実行結果 [\*]

```
/*----- matA -----*/
```

```
5.999997e+00 3.999999e+00 1.999998e+00 -5.895851e-07
7.198460e-01 8.665306e-01 2.967376e+00 3.038539e+00
-2.899769e+00 5.451697e-01 -3.095617e+00 4.233404e+00
```

```
/*----- veca0 -----*/
```

```
3.000003e+00 1.748797e+00 -5.381654e-01
```

作成したプログラムの実行結果

```
/----- 係数行列 -----/
```

```
5.999997e+00 3.999999e+00 1.999998e+00 -5.895851e-07
7.198460e-01 8.665306e-01 2.967376e+00 3.038539e+00
-2.899769e+00 5.451697e-01 -3.095617e+00 4.233404e+00
```

```
/----- 切片 -----/
```

```
3.000003 1.748797 -0.538165
```

上記の結果において、係数行列と切片が一致していることがわかる.

以上が作成したプログラムの動作確認である.

(※文責: 丹下裕太)

### 4.3.3 パターン認識

佐藤（2023）によると、マハラノビス距離に基づく多クラス線形判別とは、3クラス以上の複数のクラスから学習データが観測されたときの、各クラスの標本平均ベクトルへのマハラノビス距離を計算し、その距離の最も小さいクラスへ属すると判定する方法である [8]。以下にパターン認識のアルゴリズムを示す。

パターン認識のプログラムのアルゴリズムを数式で記述する。

マハラノビス距離に基づく多クラス線形判別を行う。

(1)  $g$  個のクラス  $i$  があるとする。各クラスのデータを次のようにおく。

$$x_{L[i]} = (x_{L[i]1}, \dots, x_{L[i]d_x})^T \quad | \quad i = 1, 2, \dots, g \quad (4.26)$$

(2) 各クラスの学習データに基づく標本平均ベクトルと標本分散共分散行列を次のようにおく。

$$m_{L[i]} = \frac{1}{N_{L[i]}} \sum_{n=1}^{N_{L[i]}} x_{L[i](n)} \quad (4.27)$$

$$S_{L[i]} = \frac{1}{N_{L[i]} - 1} \sum_{n=1}^{N_{L[i]}} (x_{L[i](n)} - m_{L[i]})(x_{L[i](n)} - m_{L[i]})^T \quad (4.28)$$

(3) 共通の標本分散共分散行列は、各クラスのデータ数でウェイトをつけて、次のように計算される。

$$S_L = \frac{1}{N_L - g} \{ (N_{L[1]} - 1)S_{L[1]} + (N_{L[2]} - 1)S_{L[2]} + \dots + (N_{L[g]} - 1)S_{L[g]} \} \quad | \quad N_L = \sum_{i=1}^g N_{L[i]} \quad (4.29)$$

(4) 各クラスの標本平均ベクトルへのマハラノビス距離を計算する。

$$d_{L(n)i} = (x_{(n)} - m_{L[i]})^T S_L^{-1} (x_{(n)} - m_{L[i]}) \quad | \quad i = 1, 2, \dots, g \quad (4.30)$$

これが最小となるクラスへと判別する。ただし、(3) は共通の標本分散共分散行列とする。

以上が、パターン認識のアルゴリズムを数式で記述したものである。

次に、作成したプログラムの動作確認を行う

デバッグ用データ 1 の実行結果 [\*]

```
/*eval count class decision matrix*/
40 0 1 0
0 50 0 0
0 0 62 0
0 0 1 40
```

## X-Reality and Smartwatch Connecting Your Feelings and the World

```
/*eval rate class decision matrix*/  
0.206186 0.000000 0.005155 0.000000  
0.000000 0.257732 0.000000 0.000000  
0.000000 0.000000 0.319588 0.000000  
0.000000 0.000000 0.005155 0.206186
```

```
/*Error rate of Eval data*/  
0.010309
```

```
/*Error num of Eval data*/  
2.000000
```

作成したプログラムでの実行結果

```
/----- eval count cla -----/  
40 0 1 0  
0 50 0 0  
0 0 62 0  
0 0 1 40
```

```
/----- eval rate cla -----/  
0.20618557 0.00000000 0.00515464 0.00000000  
0.00000000 0.25773196 0.00000000 0.00000000  
0.00000000 0.00000000 0.31958763 0.00000000  
0.00000000 0.00000000 0.00515464 0.20618557
```

```
/----- 誤判別割合 -----/  
0.010309278350515464
```

```
/----- 誤判別回数 -----/  
2
```

続いて、デバッグデータ 2 の結果を以下に示す。

```
/*eval count class decision matrix*/  
40 0 1 0  
0 50 0 0  
0 0 62 0  
0 0 1 40
```

```
/*eval rate class decision matrix*/  
0.206186 0.000000 0.005155 0.000000  
0.000000 0.257732 0.000000 0.000000
```

## X-Reality and Smartwatch Connecting Your Feelings and the World

0.000000 0.000000 0.319588 0.000000

0.000000 0.000000 0.005155 0.206186

/\*Error rate of Eval data\*/

0.010309

/\*Error num of Eval data\*/

2.000000

作成したプログラムでの実行結果を以下に示す.

/----- eval count cla -----/

40 0 1 0

0 50 0 0

0 0 62 0

0 0 1 40

/----- eval rate cla -----/

0.20618557 0.00000000 0.00515464 0.00000000

0.00000000 0.25773196 0.00000000 0.00000000

0.00000000 0.00000000 0.31958763 0.00000000

0.00000000 0.00000000 0.00515464 0.20618557

/----- 誤判別割合 -----/

0.010309278350515464

/----- 誤判別回数 -----/

2

さらに、デバッグデータ3についての結果を以下に示す.

/\*eval count class decision matrix\*/

10 0 0 0

2 8 0 0

0 0 10 0

1 0 0 9

/\*eval rate class decision matrix\*/

0.250000 0.000000 0.000000 0.000000

0.050000 0.200000 0.000000 0.000000

0.000000 0.000000 0.250000 0.000000

0.025000 0.000000 0.000000 0.225000

## X-Reality and Smartwatch Connecting Your Feelings and the World

```
/*Error rate of Eval data*/
```

```
0.075000
```

```
/*Error num of Eval data*/
```

```
3.000000
```

作成したプログラムでの実行結果を以下に示す.

```
/----- eval count cla -----/
```

```
10 0 0 0
```

```
2 8 0 0
```

```
0 0 10 0
```

```
1 0 0 9
```

```
/----- eval rate cla -----/
```

```
0.250 0.000 0.000 0.000
```

```
0.050 0.200 0.000 0.000
```

```
0.000 0.000 0.250 0.000
```

```
0.025 0.000 0.000 0.225
```

```
/----- 評価データの誤判別割合 -----/
```

```
0.075
```

```
/----- 評価データの誤判別回数 -----/
```

```
3
```

上記の結果において、係数行列と切片が一致していることがわかる.

以上が作成したプログラムの動作確認である.

(※文責: 丹下裕太)

## 第 5 章 爆弾ゲーム

この章では、3 つのグループがそれぞれ習得した技術を統合した試作品について記述する。

(※文責: 新田蒼心)

### 5.1 作成の経緯

本プロジェクトの進行に当たって、スマートウォッチ班、映像班、解析班の 3 つの班がそれぞれ学習した技術の内容を共有し、応用する手段として、成果物を作り上げる運びとなった。その際、被験者の心拍の増減を確認できるように X-Reality の技術を活用し、推移した心拍データをスマートウォッチを用いて取得、そして解析班が得た技術によるデータの解析を通じて、自分の「気持ち」の可視化を行うこととした。この可視化を達成するために、ふさわしい成果物のアイデアをメンバーそれぞれで出し合った。その話の中で出たアイデアの一部を抜粋し、以下に示す。

- 心拍数の増減に応じて天候状態を変化させ、作物を育成するゲーム
- VR 上でとあるステージ上の映像を映し出し、緊張状態をシミュレートするシステム
- VR 上に 3D モデルで作成したステージを映し出し、爆弾解除のシミュレーションを行うゲーム
- 恐怖心をあおる映像を流し、心拍の変動から恐怖の度合いを判定する診断ツール
- 音源や演出による加工を通じて、人の気分を高揚させる、あるいは落ち着かせる映像の作成

上記のアイデアのうち、得た技術を活かすという観点において、心拍の変遷をリアルタイムで検出する、という要素をわかりやすく表示するため、心拍に大きく影響を与えられる映像を作ることとなった。加えて、映像技術のうち、より感覚的に被験者に影響を与えられるであろう 3D モデルの利用が提案されたことから、没入感をより深められる VR 映像を主体にした成果物を作成する方針をとった。そして、話し合いから挙げられた上記のアイデアを採用し、最も理想的なテーマであるとして、爆弾ゲームの作成を行うこととした。素早く正確に爆弾の解体をしなければならぬ、という緊張感から、被験者の心拍数の増減がより明確にリアルタイムで現れると考えたためである。

また、自分の「気持ち」を可視化する、という目的も同時に掲げていたことから、爆弾ゲームから得られた心拍数のデータを用いて、可視化の手法をとることとした。この可視化の実装に際しては、爆弾ゲームをプレイする前後の心拍数の増減の推移から、恐怖、あるいは緊張といった感情の度合いを診断することが可能であると考えたため、感情診断の実装を加えて行うこととした。

(※文責: 新田蒼心)

### 5.2 爆弾ゲーム・感情診断の流れ

被験者は VR ゴーグル装着時は常時スマートウォッチによって心拍数が測定されており、ゲームの画面に表示されている。

1. 被験者が VR ゴーグルとスマートウォッチを装着.
2. 被験者が爆弾の解体方法を検索し発見.
3. 被験者が発見した解体方法を元に, 手持ちのコントローラーを用いて4本のワイヤーを切断.
4. 正しい順番でワイヤーを切断することができれば爆弾解体が成功となり, 爆弾のタイマーが止まる.  
誤った順番でワイヤーを切断してしまった場合は爆弾解体は失敗となり, 爆弾のタイマーが止まらなくなる.
5. 爆弾ゲーム終了. 被験者は VR ゴーグルとスマートウォッチを脱着.
6. あらかじめ用意した学習データを元に, 被験者の心拍数を用いてパターン認識を行う.
7. 被験者が爆弾ゲームを行う過程で「焦り」と「落ち着き」のどちらの感情に近かったかを調べる感情診断が完了.

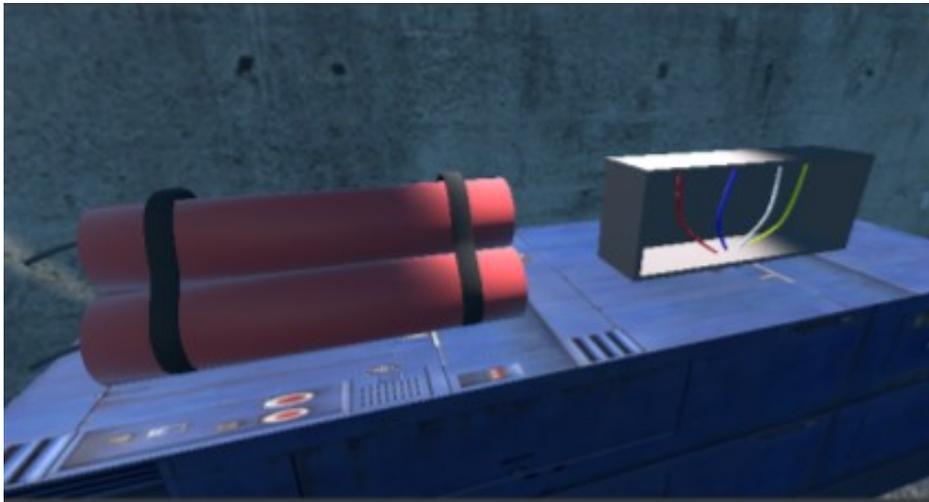


図 5.1 爆弾ゲームのプレイ画面

(※文責: 畑涼音)

### 5.3 感情診断について

解析班は, ゲームを行った際に得られた心拍数のデータを用いて感情診断を行った. 以下に感情診断の手順を示す.

1. スマートウォッチを用いて平常時の心拍数と運動をしたときの2つのパターンについて心拍数を30秒間計測した. 計測は各パターン15回行った.
2. 1で得られた心拍数のデータを csv ファイルに出力した.
3. 収集したデータを, 「焦り」, 「落ち着き」の二つのクラスに分類した.
4. 収集したデータを主成分分析を用いて4次元に圧縮し, 学習データとした.
5. ゲームを行った際に得られた心拍数のデータを主成分分析を用いて4次元に圧縮した. 以下に主成分分析のアルゴリズムを記述する.

$$x_{L_n} : n \text{ 番目の学習データ}, \quad x_L = (x_{L_1}, \dots, x_{L_{d_x}})^T : \text{平常時の心拍数と運動時の心拍数} \quad (5.1)$$

$$x_{E_n} : n \text{ 番目の評価データ}, \quad x_E = (x_{E_1}, \dots, x_{E_{d_x}})^T : \text{ゲームプレイ時の心拍数} \quad (5.2)$$

$$d_x : \text{説明変数の次元} \quad (5.3)$$

(1)  $S_L$  を求める

$$S_L : x_L \text{ の共分散行列} \quad (5.4)$$

$$S_L = \begin{pmatrix} S_{11} & \cdots & S_{1d_x} \\ \vdots & \ddots & \vdots \\ S_{d_x 1} & \cdots & S_{d_x d_x} \end{pmatrix} \quad (5.5)$$

(2)  $\lambda_S, M$  を求める

$$\lambda_S : S_L \text{ の固有値}, \quad |\lambda_{S[i]}| \geq |\lambda_{S[i+1]}|, \quad 1 \geq i \geq d_x \quad (5.6)$$

$$M : \text{主軸行列}, \quad M \equiv [e_{S_1}, \dots, e_{S_{d_x}}] \quad (5.7)$$

$$e_{S_i} : S_L \text{ の } i \text{ 番目の固有ベクトル} \quad (5.8)$$

(3)  $y_L, y_E$  を求める

$$y_L : x_L \text{ の主成分ベクトル}, \quad y_L = M^T x_L \quad (5.9)$$

$$y_E : x_E \text{ の主成分ベクトル}, \quad y_E = M^T x_E \quad (5.10)$$

以上が主成分分析を用いてデータの次元圧縮をする方法である。

6. 得られた心拍数データと学習データを用いて、マハラノビス距離によるパターン認識を行った。以下にパターン認識のアルゴリズムを記述する

(1) 2 個のクラス  $i$  がある。各クラスのデータを次のようにおく。

$$x_{L[i]} = (x_{L[i]1}, \dots, x_{L[i]d_x})^T \quad | \quad i = 1, 2, \dots, g \quad (5.11)$$

: 主成分分析後の心拍数データを学習データとした。

(2) 各クラスの学習データに基づく標本平均ベクトルと標本分散共分散行列を次のようにおく。

$$m_{L[i]} = \frac{1}{N_{L[i]}} \sum_{n=1}^{N_{L[i]}} x_{L[i](n)} \quad (5.12)$$

$$S_{L[i]} = \frac{1}{N_{L[i]} - 1} \sum_{n=1}^{N_{L[i]}} (x_{L[i](n)} - m_{L[i]})(x_{L[i](n)} - m_{L[i]})^T \quad (5.13)$$

(3) 共通の標本分散共分散行列は、各クラスのデータ数でウェイトをつけて、次のように計算される。

$$S_L = \frac{1}{N_L - g} \{ (N_{L[1]} - 1)S_{L[1]} + (N_{L[2]} - 1)S_{L[2]} + \cdots + (N_{L[g]} - 1)S_{L[g]} \} \quad | \quad N_L = \sum_{i=1}^g N_{L[i]} \quad (5.14)$$

(4) 各クラスの標本平均ベクトルへのマハラノビス距離を計算する。

$$d_{L(n)i} = (x_{(n)} - m_{L[i]})^T S_L^{-1} (x_{(n)} - m_{L[i]}) \quad | \quad i = 1, 2, \dots, g \quad (5.15)$$

これが最小となるクラスへと判別する。ただし、(3) は共通の標本分散共分散行列とする。

以下に結果の表を示す。

表 5.1 パターン認識結果

pred	pred	pred	pred	crit	attr	attr	attr
PCVec0	PCVec1	PCVec2	PCVec3	judgement class	class	class est	class judgement
8.308966	-0.30999	4.724303	0.232704	1.00E+00	sta	imp	FALSE

表のパラメータについて以下に示す。

- PCVec0-PCVec3: 主成分ベクトル
- judgement class: FALSE のとき 1, TRUE のとき 0.
- class: 事前に分類したクラス. sta: 落ち着き, imp: 焦り.
- class est: パターン認識適用後のクラス.
- class judgement: class と class est が同じなら TRUE, 違うなら FALSE を返す.

以下に結果を散布図上に可視化したものを示す。

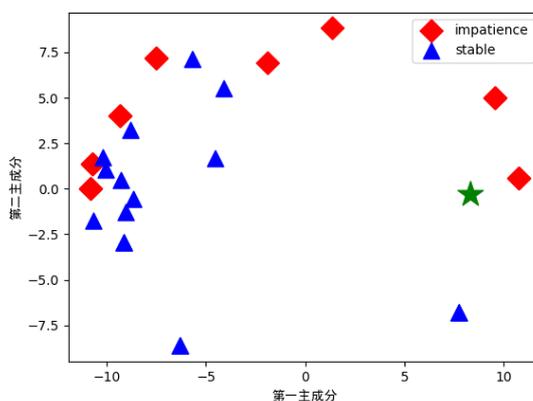


図 5.2 結果の散布図

## X-Reality and Smartwatch Connecting Your Feelings and the World

散布図上のパラメータについて以下に示す.

赤：焦りクラスの主成分ベクトル.

青：落ち着きクラスの主成分ベクトル.

緑：今回ゲームを行ったときの心拍数の主成分ベクトル.

(※文責: 丹下裕太)

## 第 6 章 中間発表会

この章では、前期に行った中間発表会 (2023,7,7) の内容に関する部分を記述する。

(※文責: 新田蒼心)

### 6.1 発表の流れ

本プロジェクトではスマートウォッチ班, 映像班, 解析班, の成果を組み合わせた爆弾ゲームについての発表を行うという形式にした。発表会の形式が前後半 (各 3 回) に分かれていたため, 各班内で発表する班員を, 前半 (品川, 結城, 川端, 高杉, 米森) と後半 (長澤, 畑, 新田, 松本, 丹下) に分けた。発表スライドは Power Point で作成し, プロジェクトの概要や活動内容をまとめたポスターをプロジェクト全体で 2 枚作成した。使用したポスターは Word で作成した。

おおまかな発表の流れは, 最初に各班のメンバが, それぞれに取得した技術を説明し, その後爆弾ゲームの実演を行った。次に取得したデータの解析の方法と実例を発表し, 最後に各班の課題を発表した。爆弾ゲームは, メンバの 1 人が実演を行い, それに合わせて映像班のメンバが 5.2 で述べられた爆弾ゲームの解説を行った。取得したデータに関する解析の方法と実例については, 使用した解析手法と使用したデータとその実例を図として提示し, 解析班のメンバが解説を行った。また, 発表スライドに QR コードを表示し, 評価フォームの記入を促した。

(※文責: 長澤颯音)

### 6.2 発表評価

この節では, 中間発表会で得られた評価内容について記述する。

(※文責: 新田蒼心)

#### 6.2.1 評価シートについて

評価フォームには, 評価者の種別や, 学籍番号または所属, 評価者指名, 発表技術の 10 段階評価, 発表技術についてのコメント, プロジェクトの 10 段階評価, プロジェクトについてのコメントの 7 項目があった。中間発表会後に集計した評価シートの回答総数は 39 件, うち 3 件が教員からの回答であった。

発表技術についての評価点は平均値が 7.79 (有効数字 3 桁), 中央値が 8 であった。

また, プロジェクトについての評価点は 7.62 (有効数字 3 桁), 中央値が 8 であった。

(※文責: 新田蒼心)

## 6.2.2 評価内容について

中間発表会を通じて得られた評価の中から、高評価であったもの、低評価であったものをそれぞれ一部抜粋し、以下にまとめる。

- 高評価（8点以上）であったコメント
  - ・スライド、ポスターが見やすい。
  - ・発表が聞き取りやすかった。
  - ・現段階の作品の完成度が高かった。
  - ・アプローチが目的に合っていたと思う。
  - ・実演の VR →後に分析の方法という感じで、流れに沿って興味を尽きさせない方法の発表がかなり良かった。
  - ・序盤に各班に分かれたことの説明があったので理解しやすかった。
- 低評価（6点以下）であったコメント
  - ・目標計画が曖昧。
  - ・XR とスマートウォッチで何を測定するのか分かりにくかった。
  - ・身体データを用いることの意味を説明して欲しい。
  - ・主成分分析などの手法の説明が欲しい。
  - ・心拍データの扱い方が分かりにくい。
  - ・グラフに用いられた詳細情報を説明して欲しい。

（※文責: 新田蒼心）

## 6.2.3 フィードバックを受けて

上記例をはじめとしたフィードバックを受けた結果についての検討、評価について以下に示す。

（※文責: 新田蒼心）

### 発表技術の評価

今回の中間発表会を通じて、発表全体に対しては、スライドやポスターの完成度の高さ、実演や説明の段取りがわかりやすかった、発表が聞き取りやすかったという評価をいただいた。これらのことから、中間発表の内容は、客観的にも充実していたと言えるだろう。一方で、身体データを用いる意義やグラフに可視化された情報の詳細が不明であったことなど、内容以前の根幹に関わる部分への説明が十分でないことが指摘された。このことから、他者へ成果を発表するにあたり、理解を促すための配慮が十分でなかったことが示された。成果発表会では内容の詳細を図や数式を使って理解を促す発表を行う必要がある。また、身体データを用いることの意義を見出していくべきである。

（※文責: 品川誠剛）

### プロジェクトの評価

プロジェクトに対しては、プロジェクト全体の技術的な分野では成果物の完成度や、目標達成に向けたアプローチの正確性などが高評価をいただくことができた。これらのことから、プロジェクト内容は、充実したものであったと言えるだろう。一方で、設定された目標が「気持ちを可視化する」だけで、具体性に欠く曖昧なものであったこと、身体データを用いる意義を説明してほしいなど、プロジェクトの根幹に関する部分への指摘があった。このことから、後期の活動では、目標の具体化を行っていく必要がある。また、身体データを用いる意義についても検討していくべきである。

(※文責: 品川誠剛)

#### 6.2.4 中間発表会後の自己評価

上記の通り、評価点は平均で7点を超過しており、発表会自体はおおむね高い評価を得られた。全体の完成度に対しても肯定的な意見が多く、私たちの活動を前面に押し出すことができたといえる。しかし、プロジェクトの前提となる用語の解説や、可視化に用いられたデータや情報の補足など、相手に理解して貰うための配慮に欠けている、という指摘も少なくなかった。これらのことから、さらに理解度を高められるような工夫を凝らす必要があることが分かった。以上を踏まえて本プロジェクトの発表は自己評価として5段階評価中4点とした。最終発表会では、技術の習得だけに目を向けるのではなく、メンバ全員が今回の反省を活かして、より良い評価を得られるように努めていきたい。

(※文責: 品川誠剛)

## 第7章 まとめ

本プロジェクトは、「X-Reality とスマートウォッチを用いて身の回りの現象に接続し、自分の気持ちと世界をつなぐための情報リンクを作成する」ことを目的とするプロジェクトである。その目的を達成するために、前期ではメンバを映像班、スマートウォッチ班、解析班の3つの班に分け、目的の達成を成し得るために必要な技術の習得を主な目標として活動した。

映像班は、Unity Editor の使用方法の理解、360度カメラで撮影した映像を Unity 上で 3D モデルとして生成する手法の実現、それらを VR 上に連動させて動作させる技術を習得した。また、成果物である爆弾ゲームの制作に際して、フリーのテクスチャや自作した爆弾の 3D モデルなどを設置し、ゲームにおけるステージの作成も行った。

スマートウォッチ班は、スマートウォッチで計測できる身体データを Node.js と通信規格の 1 つである ANT+ 通信を使用することでリアルタイムでの計測、および表示を行う技術を習得した。また、計測した身体データを csv ファイルとして PC に保存する技術についても学習した。

解析班は、スマートウォッチ班が取得したデータを、統計学的手法を用いて解析するため、主成分分析、重回帰分析、マハラノビス距離を用いたパターン認識の3つの解析手法それぞれのアルゴリズムを学習した。また、成果物の感情診断を実装するに当たって、計測したデータに対して主成分分析を行いデータの次元を圧縮し、そのデータにパターン認識を行うことで診断結果の可視化を行った。

その後、それぞれの班が習得した技術を統合し、目的に沿った成果物の試作品である「爆弾ゲーム」と「感情診断」を制作、および実施した。

この章では、本プロジェクトの前期における活動内容や今後の改善点を述べる。

(※文責: 新田蒼心)

### 7.1 各グループのまとめ

この節では、前期における各グループの活動を述べる。また、今後の課題点も挙げる。

(※文責: 新田蒼心)

#### 7.1.1 映像班のまとめ

本グループは前期を通して、「Virtual Reality」や「Augmented Reality」についての技術の習得と成果物の作成に取り組んだ。その甲斐もあり、中間発表会では動作する成果物を発表することができた。しかし、習得した技術の中には、未熟なまま習得期間が過ぎてしまったものもあり、前期の成果物に取り込むことができなかった。また、メンバ間での技術の習得度合いがずれている節もあり、特定のメンバが手持ち無沙汰になることもあった。

(※文責: 結城熙乙)

### 7.1.2 スマートウォッチ班のまとめ

本グループは前期を通して、スマートウォッチと Node.js を使用し、通信規格の一種である ANT+通信を介することで、PC に心拍数をリアルタイムで送信し計測する手法の習得、および送信されたデータを csv ファイルに保存する技術の習得と、成果物である「爆弾ゲーム」の作成に取り組んだ。しかし、得られた身体データが心拍数のみであったことにより、感情診断を行う際に必要な学習データの種類が不足してしまった。その結果、感情診断で導き出された可視化結果が、信憑性の欠けるものとなってしまった。

(※文責: 新田蒼心)

### 7.1.3 解析班のまとめ

本グループは前期を通して、得られた身体データを活用するために、主成分分析、重回帰分析、パターン認識の 3 つの統計学的手法についての学習と成果物の作成に取り組んだ。中間発表会で、動作する物を作成し、感情診断を形にすることはできたが、各手法に対して、メンバー間で理解度の差が生まれてしまった。

(※文責: 丹下裕太)

## 7.2 反省と今後の展望

この節では、メンバーそれぞれが前期の活動を通して見つけた反省点と改善策について記述する。

(※文責: 新田蒼心)

### 7.2.1 各メンバーの反省

メンバーそれぞれが見つけたプロジェクト進行に関する反省点について以下にまとめる。

(※文責: 新田蒼心)

スケジュールの管理や情報の共有が不十分だったために、プロジェクト自体をうまく機能させることができず、必要以上に時間を浪費してしまった。

(※文責: 川端勇雅)

全体スケジュールの管理不足により、中間発表会の準備に取りかかるタイミングが遅くなってしまい、時間が足りずに発表内容を完全なものにしきれなかった。

(※文責: 品川誠剛)

スケジュールの管理不足により、グループごとの作業の進行度にずれが生じてしまい、手持ち無沙汰になってしまうメンバーがいる時間ができてしまった。

(※文責: 松本柊哉)

## X-Reality and Smartwatch Connecting Your Feelings and the World

グループ間やメンバ間で情報共有が足りておらず、進捗や習得した技術の認識にずれがあった。そのため、すりあわせの時間を捻出せざるを得なくなってしまった。

(※文責: 結城熙乙)

グループ間の技術、知識などの情報共有を行う時間が十分でなかったため、作業が思うように進められなかった。

(※文責: 米森麻翔)

メンバ間で情報の共有が不十分だったため、技術的な問題に気が付けず、先生からアドバイスをもらうのが遅れてしまった。

(※文責: 高杉拓海)

班の間での情報共有が足りておらず、成果物作成のスケジュールに遅れが生じてしまった。

(※文責: 丹下裕太)

前期の期間を念頭に置かずに目標を高く設定してしまったことで、期日までにすべての作業を間に合わせることができなかった。

(※文責: 長澤颯音)

情報の共有が遅れてしまったことにより、他班の進捗を知るタイミングが集まった時間にしかとれなくなってしまい、成果物の制作アイデアを出すのに必要以上に時間を要してしまう結果となった。

(※文責: 新田蒼心)

分担作業の理解がメンバそれぞれに行き届いていなかったことで、手持ち無沙汰になるメンバがいるときができてしまい、それに周りも気づかないことがあった。

(※文責: 畑涼音)

### 7.2.2 今後の展望

メンバそれぞれが挙げた反省点に対し、今後の活動で意識するべきだと考えた改善点について以下にまとめる。

(※文責: 新田蒼心)

時間配分や作業効率を向上させるため反省点のフィードバックを入念に行い、プロジェクトをうまく機能させる。

(※文責: 川端勇雅)

全体的な活動に遅れを生じさせないために、スケジュールをあらかじめ立てて、見通しをメンバ全員で周知した上で行動に移す。

(※文責: 品川誠剛)

## X-Reality and Smartwatch Connecting Your Feelings and the World

プロジェクト開始時にスケジュールを設定して作業に移り、状況に応じてスケジュールを見直し、変更する意識を持つ。

(※文責: 松本柊哉)

プロジェクト全体で情報共有する時間をあらかじめ決めておくなどして、認識のずれを未然に防ぐ手立てを立てておく。

(※文責: 結城熙乙)

毎回、冒頭で教員に対して行う進捗の報告時間を利用して、グループ間の交流も積極的に行い、より詳細な進捗報告を行う時間を作る。

(※文責: 米森麻翔)

解釈や目的が個人によって異なるように、曖昧な表現は使わないようにし、議論内容を具体的にする。

他の班に情報共有をする際には、同じ班のメンバによって説明の仕方が異なるようにする。

(※文責: 高杉拓海)

班の間で何を行ったのか、何が問題になっているのかなどを定期的に共有し、優先事項を誤らないように徹底する。

(※文責: 丹下裕太)

必要な機能を分析し、そこから期日内に間に合うかどうか逆算した上で、機能の取捨選択や目標の設定を行う。

(※文責: 長澤颯音)

毎回の活動の終わりを各班に任せるのではなく、進捗の共有を行う時間を設け、他班の進み具合も把握しておく体制を整えるよう徹底する。

(※文責: 新田蒼心)

その日の始まりに誰が何の作業を行うのか、どれくらい時間がかかるかなどの情報を紙などに書き出し、明確化してメンバ同士で共有しておく。

(※文責: 畑涼音)

## 第 II 部

### 後期

## 第 8 章 はじめに

この章では、後期における本プロジェクトの目的やグループごとの概要について述べる。

(※文責: 結城熙乙)

### 8.1 メンバの編成

前期では、メンバを3つのグループに分け、それぞれの技術習得と成果物作成までの一連の流れを行った。

後期では、グループを再編し、前期で得た技術を用いた成果物をそれぞれ3つ作成した。以下 A 班, B 班, C 班とする。

後期の活動におけるグループのメンバの内訳は以下の通りである。

表 8.1 後期グループとメンバの内訳

担当/班	A 班	B 班	C 班
解析	米森	高杉	丹下
SW	新田	松本	川端
映像	結城	品川	長澤

(※文責: 新田蒼心)

### 8.2 プロジェクトリーダー

後期では、プロジェクトリーダーを交代制とし、2週間で交代することとした。内訳は以下のとおりである。

1. 米森
2. 丹下
3. 長澤
4. 高杉
5. 松本
6. 結城

以降は1から繰り返す。

(※文責: 川端勇雅)

## 8.3 グループ概要

この章では、各グループの概要と目標について記述する。

(※文責: 結城熙乙)

### 8.3.1 A 班

運動データを収集し、そのデータを活用して「楽しさ」を可視化することで楽しく運動してもらうことを目的とした。目標達成のため、3D モデルで街並みを再現し、それを見ながらトレッドミルを歩いてもらい、その際にスマートウォッチより得られた運動データを解析することでユーザーの楽しさの度合いを推測するシステムの開発を行った。

(※文責: 米森麻翔)

### 8.3.2 B 班

B 班は、本プロジェクトの目的である「X-Reality とスマートウォッチを用いて身の回りの現象に接続し、自分の気持ちと世界をつなぐための情報リンクを作成する」ことに沿った成果物の制作を行うことを目的とした。目的達成のため、VR ゴーグルで恐怖症映像を視聴した際の心拍数データを解析し、結果からユーザーが恐怖症か否かを診断するシステムの開発を行った。

(※文責: 高杉拓海)

### 8.3.3 C 班

リアルタイムに反映させた心拍数データによって睡眠を補助するアプリケーションを開発を行った。具体的には、睡眠の促進、目覚めの改善、睡眠の見直しを行うことを目標とした。目的達成のため、睡眠中の心拍数の計測、心拍数が一定の値を上回ったときに鳴るアラーム、心拍数をまとめた CSV ファイルの出力、それを用いたパターン認識での睡眠傾向の判別などを行った。パターン認識では事前に学習データを取る必要があったが、時間不足で十分なデータを取ることができず、あまり信ぴょう性があるとは言えない結果になってしまった。

(※文責: 長澤颯音)

## 8.4 活動計画

この章では、各グループの後期の活動計画について記述する。

(※文責: 品川誠剛)

### 8.4.1 A 班

- 9 月
  - ・後期製作物の内容を決定するにあたって、各メンバー3つずつ案を持ち寄り、それぞれの案を組み合わせて「おうち世界旅行」を製作することに決定した。
  - ・作業の環境は前期に用いた PC を引き継いで使用することとなった。
- 10 月
  - ・重回帰分析のプログラム作成。
  - ・運動データの収集, 学習データの作成。
  - ・使用する 3D モデルの選定を行い、札幌市狸小路ステージは PLATEAU の 3D モデルを使用することに決定。
  - ・Unity にてステージの作成を開始。
- 11 月
  - ・運動データの収集, 学習データの作成。
  - ・Unity でのステージ制作。
  - ・プロトタイプの発表。
- 12 月
  - ・プロトタイプ発表の反省
  - ・各ステージの挙動の調整
  - ・成果発表会
  - ・フィードバックからの反省
  - ・グループ報告書作成
- 1 月
  - ・グループ報告書の加筆修正
  - ・学習ポートフォリオ, 学習フィードバックシート, プロジェクト報告書の作成。

(※文責: 品川誠剛)

### 8.4.2 B 班

- 9 月
  - ・後期に取り組む製作物を決めるために案出しを行い、班ごとに3つの候補を挙げて、実行の容易性、面白さ等の観点から主観で数値化し、必要な映像や一連の流れを記した草案を作成した。
- 10 月
  - ・草案と数値を基にして製作物を「恐怖症診断」に決定。
  - ・VR ゴーグルで恐怖症映像を見たときの心拍数データからユーザの恐怖の対象を診断するシステムの開発を目標とした。
  - ・高所, 深海, 先端恐怖症に対応した映像の作成, 作成のため 360 度カメラ, Unity, Blender の使用を決定。
  - ・心拍数データの計測手法をスマートウォッチに, 解析手法を, 主成分分析, パターン認識に決定。

## X-Reality and Smartwatch Connecting Your Feelings and the World

- ・新規 PC の初期設定, Discord, Unity, Oculus アプリ, Visual Studio Code のインストールを行った.
- ・使用する恐怖症映像の撮影.
- ・プロトタイプ発表会に向けて恐怖症映像視聴時の心拍数データの収集, 学習データの作成を行った.
- ・主成分分析とパターン認識を行うプログラムで学習データの解析を行った.
- 11 月
  - ・高所恐怖症に対応した映像の撮影.
  - ・Unity を SW と Oculus に連動させる.
  - ・平常時の心拍数を取得するプログラムを書くなどのシステムを作成.
  - ・プロトタイプ発表会.
  - ・Unity, Blender を用いて深海, 先端恐怖症に対応した映像を作成した.
  - ・作成した映像を VR ゴーグルで視聴するためのアプリケーションを作成した.
  - ・プロトタイプ発表会のフィードバックを受けて, 映像視聴時の心拍数データの計測方法とデータの前処理の検討を行った.
  - ・アプリケーションを用いて, 恐怖症映像視聴時の心拍数データの収集, 学習データの作成を行った.
  - ・移動平均フィルタ, 主成分分析, パターン認識を行うプログラムを用いて学習データの解析を行った.
  - ・成果発表会に向けた, サブポスター, 発表スライドの作成を行った. ポスター作成にあたり, 解析結果の可視化を行った.
- 12 月
  - ・成果発表会
  - ・主成分分析で圧縮する次元数を変更して学習データの解析を行った.
  - ・グループ報告書の作成.
- 1 月
  - ・グループ報告書の修正.
  - ・学習ポートフォリオ, 学習フィードバックシート, プロジェクト報告書の作成.

(※文責: 品川誠剛)

### 8.4.3 C 班

- 9 月
  - ・後期に取り組む成果物の案を 3 つ考えた.
    - － 運動をサポートするアプリケーション.
    - － 消費カロリーに応じて敵を倒すゲーム.
    - － 睡眠促進 VR.
  - ・以上 3 点から話し合い, 成果物を「睡眠促進 VR」に決定した.
  - ・スマートウォッチの選定.
  - ・今後やることの話し合い.
  - ・グループリーダー決め.

## X-Reality and Smartwatch Connecting Your Feelings and the World

- 10月
  - ・ Unity の復習, アプリ開発開始.
  - ・ ポスター作成開始.
  - ・ 解析に適した睡眠時心拍数の csv ファイルの加工.
  - ・ Unity に心拍数を送る機能の実装.
- 11月
  - ・ 先月に引き続きアプリ開発.
  - ・ 睡眠データの収集, 学習データの作成を行った.
  - ・ 主成分分析とパターン認識で学習データの解析を行った.
  - ・ 成果物のプロトタイプ発表.
  - ・ プロトタイプの改善. データ解析方法の検討を行った.
  - ・ アプリのデザインを調整, 背景を付けた.
  - ・ 成果発表会に向けてポスター, スライドや動画の準備.
- 12月
  - ・ アプリの完成.
  - ・ 成果発表会の本番.
  - ・ 発表評価の確認, 反省.
  - ・ 期末提出物の準備.
- 1月
  - ・ 報告書の修正, 提出.
  - ・ 学習ポートフォリオ, 学習フィードバックシート, プロジェクト報告書の作成.

(※文責: 品川誠剛)

## 第9章 A班

この章では、A班の後期の活動内容について述べる。

(※文責: 新田蒼心)

### 9.1 テーマの決定

A班では、前期の活動を踏まえて、スマートウォッチや X-Reality を用いて、身体データを取得し、それらのデータを活用し解析することで、自分の「気持ち」を可視化する。という目的を掲げた。その目的を達成するために、得られた技術を用いたデータの収集と、活用したデータの可視化を行うための手法についてのアイデアを、メンバそれぞれで出し合った。その話し合いで出たアイデアの抜粋を以下に示す。

- 運動量に応じてランクを分け、それに応じてアバターを育成するゲーム
- VR 上で観光スポットの映像を共有し、擬似的に観光ができるシステム
- スマートウォッチで実際の運動量を計測し、他者と比較・共有できるシステム

これらのアイデアの中から、まず、スマートウォッチから得られる身体データの中で、最も多く挙げられた運動データを利用して、可視化を行う案に着目した。それに加えて、自分の「気持ち」を可視化するという目的を達成するため、「気持ち」の中で、「楽しさ」の感情に焦点を当て、気分を盛り上げられるような映像を使用する、という案が出た。そこで、疑似観光システムの映像案に白羽の矢が立てられ、運動データを取得する案と同時採用し、両者を組み合わせた成果物を作成する運びとなった。その際、映像を見ながらの運動になるということから、体を激しく動かす運動との併用が難しいと考えられたため、トレッドミルを使用し、安全性を担保することとした。そして、トレッドミル上での歩行運動を散策に見立て、より現実的な疑似観光を行うことができる折衷案を立てた。それらのアイデアを踏まえて、3Dモデルで実際の町並みを映像上にて再現し、その映像を眺めながらトレッドミル上を歩くことで運動データを集め、収集したデータを活用してユーザー自らの「楽しさ」を可視化するシステムを作り上げることに決定した。

(※文責: 新田蒼心)

### 9.2 成果物紹介

この節では、後期の活動における、A班の成果物について述べる。

A班の成果物のタイトルは「おうち世界旅行」であり、Windows PC 上で動作する運動ゲームである。スマートウォッチを腕に装着したうえでゲームを起動し、画面の前で歩行運動をすることで仮想空間を散策することができるというものである。このとき、トレッドミルを用いてもよい。また、ゲームに登場する仮想空間は実在する都市をモデルにしたものである。執筆時点で実装できていたものについては、札幌市狸小路と、東京都秋葉原をモデルにした仮想空間である。また、本ゲームのプレイ中(仮想空間内の散策)に生じた運動データの収集、および解析を行い、運動の楽

しさが何と深い関係があるのかを確かめる。

(※文責: 結城熙乙)

アプリケーションの流れを図にまとめたものを以下に示す。

1. 都市の選択

- 3Dモデルで再現された都市
- 映像をスクリーンに表示



2. トレッドミル上で都市を疑似散策

- SWを身に着け、計測を行う
- 映像を見ながら実際に歩く



3. SWから得られた運動データを入力

- 時間、ペース、最大心拍数を計測

4. データを解析して運動の「楽しさ」を出力

- 重回帰分析を用いてユーザの楽しさを予測



図 9.1 アプリケーションの流れ

(※文責: 結城熙乙)

## 9.2.1 映像技術

ゲームの映像の制作にあたっては、Unity を用いた。ゲーム内に登場する都市モデルの入手元は、以下に示す通りである。

- 株式会社ゼンリン: ZENRIN ASSET Series
  - Unity Asset Store にて入手可能
- 国土交通省主導: Project PLATEAU
  - G 空間情報センター にて入手可能

(※文責: 結城熙乙)

## 9.2.2 プレイの手順

1. スマートウォッチを腕に装着し、心拍送信モードにする。
2. PC にて「おうち世界旅行」ランチャを起動後、プレイする都市をクリックし選択する。
3. 一人称視点の映像と、リアルタイムの心拍数の表示が始まるので、腕に装着した心拍数と相違ないことを確認する。
4. 画面の前で歩行運動 (足踏み) することで、心拍数が上昇する。このとき、トレッドミルを用いてもよい。

5. 心拍数が上昇すると、映像の移動速度も比例して上昇する。

(※文責: 結城熙乙)

### 9.2.3 解析の流れ

1. 学習データの作成.
2. 学習データを正規化.
3. 正規化した学習データに対して重回帰分析を行う.
4. 重回帰分析より得られた回帰係数を用いて被験者の楽しさを評価する.

(※文責: 米森麻翔)

### 9.2.4 解析技術

被験者に映像を見ながら歩いてもらい、その際にスマートウォッチやトレッドミルより得られたデータに対して重回帰分析を行い、被験者の楽しさを5段階評価で解析した。重回帰分析における説明変数には「運動時間」「運動時の最大心拍数」「ウォーキングマシンのペース(時速)」「目標心拍数と最大心拍数の差」とした。目標心拍数はカルボーネン法(後述)を用いて求めた。

(※文責: 米森麻翔)

#### 学習データ

重回帰分析を行うにあたって、学習データは被験者にインターネット上にある観光映像を見ながらスマートウォッチを装着して実際にトレッドミルを歩いてもらい、運動後に楽しさを1~5の5段階で評価してもらった。その際、運動時間やトレッドミルのペースを変更しながら計測し、各条件下での楽しさの評価を比較した。しかし、一部データについては上記の取得方法では十分なデータ量が確保できなかったため、研究者の主観的な判断に基づいて生成されたダミーデータが含まれている。これらのダミーデータについては、実データと同様の特性を模倣するように生成されたが、その信頼性に関しては注意が必要である。

(※文責: 米森麻翔)

#### カルボーネン法 [11]

カルボーネン法とは、心拍数や運動強度を用いて運動時の目標心拍数を求めるための方法である。具体的な計算式は以下の通りである。

$$\text{目標心拍数} = (\text{最大心拍数} - \text{安静時心拍数}) * \text{運動強度} + \text{安静時心拍数} \quad (9.1)$$

$$\text{最大心拍数} = 220 - \text{年齢} \quad (9.2)$$

運動強度は0から1の間で設定するが、本システムではウォーキングを想定している。ウォーキングの運動強度として0.5を使用することが一般的であるため今回は上式の運動強度には0.5を使用した。

(※文責: 米森麻翔)

## データの解析

作成した学習データに対して、変数の偏りを無くすために学習データを正規化した。正規化した学習データに重回帰分析を行った。重回帰分析のアルゴリズムについては第4章3.2節を参照。以下に解析結果を示す。

表 9.1 回帰係数

Column	Coefficient
time	0.577193
HB_Max	- 0.268433
pace	- 0.313558
HB_Diff	- 0.502547

上記の表は重回帰分析を行った結果得られた各説明変数の回帰係数である。説明変数について、time が運動時間、HB\_Max が運動時最大心拍数、pace がトレッドミルのペース、HB\_Diff が目標心拍数と最大心拍数の差である。表より楽しさにおいて time が最も大きく寄与しており、時間が長いほうが楽しく感じやすい傾向にあることがわかった。続いて HB\_Diff が楽しさに寄与しており目標心拍数と最大心拍数の差が小さい、すなわち最大心拍数が目標心拍数に近いほど楽しく感じる傾向にあることがわかった。

(※文責: 米森麻翔)

## 9.3 使用した技術

(ANT+, 3D モデル, Node.js)

(※文責: 結城熙乙)

## 9.4 仕様書

### 9.4.1 ソフトウェア名

おうち世界旅行

### 9.4.2 開発環境

- OS: Windows 11 Home
- IDE: Unity Editor 2021.3.25.f1
- エディタ: Sublime Text 4143

### 9.4.3 動作環境

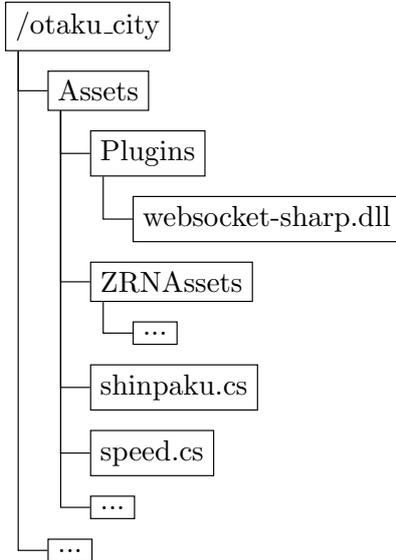
- Windows (x84-64 かつ Windows10 以降, Windows11 が望ましい.)

- Node.js v18.16.8

#### 9.4.4 ディレクトリ構成とソースコード

本ソフトウェアのディレクトリ構成は以下のとおりである。一部のファイルは記載を省略している。

以下に、秋葉原ステージのディレクトリ構成を示す。札幌市狸小路ステージは概ね同様の構成をしているのでそちらに関しては省略する。



- Unity で WebSocket 通信をするために、WebSocketSharp というプラグインをビルドして導入している。
- shinpaku.cs において、Node.js サーバと WebSocket 通信をしてリアルタイム心拍数データを取得する処理をしている。また、main 関数がこのスクリプトにある関係で、esc キー押下でゲーム終了する処理もこの中に記述した。
- speed.cs には、取得した心拍数データに基づいてプレイヤーの移動速度を変化させる処理を記述している。

次のセクションに、speed.cs と shinpaku.cs の各ソースコードと解説を行う。

(※文責: 結城熙乙)

#### スクリプト部のソースコードと解説

以下が shinpaku.cs のソースコードである。

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using WebSocketSharp;
using UnityEngine.UI;
```

```
public class shinpaku : MonoBehaviour
{
    // Start is called before the first frame update
    private WebSocket ws;
    public static string stackText;
    private Text targetText;

    void Start ()
    {

        var url = "ws://localhost:8080";
        ws = new WebSocket(url);
        ws.Connect();

    }

    // Update is called once per frame
    void Update()
    {

        EndGame();

        ws.OnMessage += (sender, e) =>
        {
            stackText = e.Data;
        };

        this.targetText = this.GetComponent<Text>();
        this.targetText.text = "現在の心拍数:" + stackText.ToString();
    }

    void EndGame(){
        if (Input.GetKey(KeyCode.Escape))
        {
            Debug.Log("aa");

            #if UNITY_EDITOR
                UnityEditor.EditorApplication.isPlaying = falseゲーム
プレイ終了;//
            #else
```

```
        Application.Quitゲームプレイ終了();//
    #endif
    }
}
}
```

以下が speed.cs のソースコードである。

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using WebSocketSharp;
using UnityEngine.UI;
using Cinemachine;
using System;

public class speed : MonoBehaviour
{
    // Start is called before the first frame update
    private WebSocket ws;
    private string stackText;
    private Text targetText;

    CinemachineDollyCart myDolly;
    float dollySpeed = 0;
    String tempText;

    private void Awake(){
        myDolly = GetComponent<CinemachineDollyCart>();
    }

    // Update is called once per frame
    void Update()
    {
        tempText = shinpaku.stackText;
        myDolly.m_Speed = Convert.ToSingle(tempText) * 0.030f;

        /*
        switch (Convert.ToInt32(tempText) / 10){
            case 7: myDolly.m_Speed = 0.0f; break;
            case 8: myDolly.m_Speed = 2.0f; break;
        }
        */
    }
}
```

```
        case 9: myDolly.m.Speed = 4.0 f; break;
        case 10: myDolly.m.Speed = 5.5 f; break;
        case 11: myDolly.m.Speed = 9.0 f; break;
        case 12: myDolly.m.Speed = 11.5 f; break;

        default: break;
    }
    */

    //dollySpead = Convert.ToSingle(tempText) * 0.01 f;
    //myDolly.m.Speed = dollySpead;

}
}
```

プレイヤーの移動の表現については、Unity の機能の一つである Cinemachine を用いた。本ゲームにおいては、プレイヤーカメラを Cinemachine dollyCart にアタッチしたうえで、指定したパス（ルート）を自動的に移動させることで実現している。この dollyCart オブジェクトの速度 myDolly.m.speed を、「心拍数 / 10」の値に応じて、乗数を変更することで段階的な速度の変化を実現している。また、下のほうでコメントアウトされているように、心拍データを数値に変換したものに、0.01 を掛ける方法もあったが、変化が分かりにくいため没案となった。

(※文責: 結城熙乙)

#### 9.4.5 各ステージの詳細な解説

ここでは、各ステージのモデルの出典や、インポートの際の手順について述べる。

(※文責: 結城熙乙)

##### 札幌市狸小路ステージ

「おうち散歩」の札幌市狸小路ステージに関しては、国土交通省が主導する、日本全国の 3D 都市モデルの整備・オープンデータ化プロジェクト「PLATEAU」にて配布されている、OBJ/FBX 形式の 3D モデルを用いて作成した。Unity プロジェクトにインポートするための手順は以下に示すとおりである。

- G 空間情報センター / 3D 都市モデル (Project PLATEAU) 札幌市 (2020 年度) (<https://www.geospatial.jp/ckan/dataset/plateau-01100-sapporo-shi-2020>) より、任意の形式の 3D 都市モデルをダウンロードする。
- Unity Hub で、3D 都市モデルをインポートしたいプロジェクトを開く。
- Unity Editor 上の、プロジェクトビューの Assets にインポートしたいファイルをドラッグアンドドロップする。
- ヒエラルキー上にインポートしたモデルが出現するので、座標や向き、スケールを適切に設

定する。

(※文責: 結城熙乙)

### 秋葉原ステージ

「おうち散歩」の秋葉原ステージに関しては、株式会社ゼンリンがフリーで提供している、(MIT ライセンス) ZENRIN City Asset Series の「Japanese Otaku City」を用いて作成した。Unity Asset Store より、Unity プロジェクトにインポートすれば即時利用可能な状態で頒布されているため、札幌市狸小路ステージよりは制作が円滑に進んだと感じた。

Unity プロジェクトへのインポートの手順については、前述のとおり Asset Store より入手する方式のため割愛する。

(※文責: 結城熙乙)

### 9.4.6 解析

本ソフトウェアにおける解析の仕様について、以下に示す。

#### 動作環境

- Windows 10 Pro
- Python 3 (Jupyter Notebook)

(※文責: 米森麻翔)

## 9.5 まとめ

ここでは A 班が行った活動のまとめと反省点について述べる。

A 班は、スマートウォッチと X-Reality を活用し、得られた身体データから自分の「気持ち」を可視化するという目的を掲げている。その目的を達成するため、取得する身体データの内から運動データ、X-Reality からは映像技術の 1 つである 3 D モデルに焦点を当て、3 D モデルで作成した町並みの映像を眺めながらトレッドミル上を実際に歩き、疑似観光を行えるシステムを作成した。さらに、その運動で発生したデータを収集し、重回帰分析をすることで、ユーザーの「楽しさ」の度合いを 5 段階で数値化して可視化する実験を行った。

実験の末、運動データ(時間)と「楽しさ」の度合いには、比例的な傾向がある可能性を見取ることができた。

その一方で、歩くペースの早さと「楽しさ」の度合いには反比例的な傾向を見ることができた。

このことから、ゆったりと長めの運動を行う方が他の条件に比べて楽しめる可能性があることがわかった。

ただし、班員を被験者として実験を行ったため、データの数が十分ではなかったことから、結果の信憑性には欠ける点を留意しなければならない。

(※文責: 新田蒼心)

### 9.5.1 映像担当のまとめ

映像班においては、仮想空間の実現にあたって、前期で学習した Unity の技術を用いて制作を行った。また、スマートウォッチからリアルタイムの心拍データを受け取る部分の実装においては、C#のスクリプトを用意して実際に動くところまでの実装を行った。各種開発のスケジュールについては余裕のあったものの、報告書を作成するための前準備が不足しており、結局後半に余裕が無くなってしまった。そのため報告書のことを考えたタスク配分を行う必要があったと考えられることが、映像班の反省点であったといえる。

(※文責: 結城熙乙)

### 9.5.2 スマートウォッチ担当のまとめ

運動データを可視化に利用するため、歩行運動時の時間、ペース、ユーザーの最大心拍数を計測した。加えて、システム画面上に心拍数の変遷をリアルタイムで表示するため、Node.js を経由し pc にデータを送信した。前期の活動で得られた技術を主に用いたため、作業において滞ることは少なかったと考えている。ただし、他班の技術との併用をするにあたり、他班の技術の進捗や概要を把握しきれておらず、実装を班員に委ねることが多々あった。そのため、技術関連の共有を怠ってしまったことで、実装に費やす時間が見積もりよりも多くなってしまったことが反省点であると考えられた。

(※文責: 新田蒼心)

### 9.5.3 解析担当のまとめ

解析班では、映像を見ながら運動を行った際に得られるデータの計測解析を、解析を行った。データ取得にはスマートウォッチやトレッドミルを用いた。これらのデータに重回帰分析を行って解析することにより被験者の運動に対する楽しさを推測することができた。しかし、解析に用いる学習データ量が少なく完全な解析には至らなかった。そのため反省点としては、テーマと解析手法を早期に決定しデータ取得に取り組むべきであった。

(※文責: 米森麻翔)

### 9.5.4 各メンバーの反省

映像の制作においては自分が中心となって行った。自分と各班との連携が不足していたのは、自分が自発的に連絡を取れていなかったところが大きいと思う。対して、他のメンバーが自発的に声掛け等を行ってくれたので、助かった点は大きかった。

(※文責: 結城熙乙)

成果物の方向性を定めるための話し合いに多くの時間を費やしすぎた。そのため、全体のスケジュールに遅れが生じてしまい、内容の充実度合いや得られた結果の信憑性が満足いく水準にまで到達できなかったと考えている。グループリーダーとして、常に締め切りを念頭に置いたタスクの

## X-Reality and Smartwatch Connecting Your Feelings and the World

進行を心がけるべきであり、そこが最大の反省点であったと考える。

(※文責: 新田蒼心)

解析手法の決定とデータ取得を始めるタイミングが遅く、満足な解析結果を得ることができなかった。また、解析を行うプログラム作成にも時間がかかってしまったため、不明点は早期に教員やメンバに相談するべきであった。

(※文責: 米森麻翔)

## 第 10 章 B 班

### 10.1 目的

B 班は、本プロジェクトの目的である「X-Reality とスマートウォッチを用いて身の回りの現象に接続し、自分の気持ちと世界をつなぐための情報リンクを作成する」ことに沿った成果物の制作を行うことを目的とした。

また、後期からは、前期に振り分けた映像班、SW 班、解析班を一名ずつ集めて 1 つの班として、1 つの班につき 1 つの成果物を作った。

(※文責: 品川誠剛)

### 10.2 テーマの決定

テーマの決定にあたって、目的に沿う案を班員各自で持ち寄り、「VR 観光」「恐怖症診断」「作物育成ゲーム」の 3 つを案として捻出し、その中から「面白さ」「使用頻度」「実行の容易性」の 3 つの観点から「恐怖症診断」を制作することに決定した。

恐怖症診断をやる上での懸念点として、VR の映像を見るだけで心拍数が上がるのかというのがあったので、恐怖映像だけで心拍数が上がるのかを確認し、問題ないと判断して恐怖症診断の制作に踏み切った。

(※文責: 品川誠剛)

### 10.3 成果物

恐怖症診断の実現のため、前期に各メンバが学んだ技術を用いて、ユーザがどの恐怖症であるか否かを診断するシステムの開発を目標とした。まず、システムの基盤となる恐怖症映像の作成を行った。高所、深海、先端という限局性恐怖症に含まれる事象、ものを対象とした恐怖症映像を VR ゴーグルに投影するアプリケーションを開発した。アプリケーションで恐怖症映像を視聴した際の心拍数データをスマートウォッチで計測し、解析プログラムを用いて計測結果を解析し、ユーザが各恐怖症映像に対して、どの程度恐怖を感じているかを推測するシステムを開発した。以下の図 10. 1 がこの成果物のシステム構成図である。以下にアプリケーション、計測に使用したシステム、解析プログラムの詳細について説明する。



図 10.1 システムの構成図

### 10.3.1 心拍数受信・書き出しシステム

スマートウォッチからパソコンに心拍数を取り込むシステムは、第3章で使用した ant-plus のプログラムを改良した。本グループは、VR ゴーグルを同時に使用し、PC の負荷が高くなる可能性が考えられたため、3.4.2 で述べられている書き出しのプログラムは使用しなかった。

#### 使用方法

1. 装着したスマートウォッチを心拍送信モードにする。
2. StartHeartBeats.bat を起動する。
3. cmd 上に心拍数が表示されていれば正常に起動しており、同ディレクトリ内に心拍データが CSV として出力される。

(※文責: 松本柊哉)

### 10.3.2 アプリケーション

本アプリケーションは、VR ゴーグルと PC を接続し、アプリケーションを起動すると恐怖症に対応する映像が再生される。実装した映像は高所恐怖症、深海恐怖症、先端恐怖症の3つである。高所恐怖症の映像は、360度カメラで撮影したもので、深海恐怖症、先端恐怖症の映像は3Dモデルで再現された映像である。

アプリケーションは3つのシーンに分けられる。第1シーンはプレイヤーの平常時の心拍を計測するシーンであり、アプリケーション起動時に、真っ暗な映像が流れる。第2シーンは恐怖症の映像が再生される。第3シーンは恐怖症映像視聴後の心拍数の変化を計測するシーンであり、恐怖症映像の間に再生され、真っ暗な映像が流れる。以下にアプリケーションの流れを記述する。また、図 10.2、図 10.3、図 10.4 は各映像のスクリーンショットである。

#### アプリケーションの流れ

1. アプリケーションを起動すると、真っ暗な映像が 30 秒間流れる。
2. 高所恐怖症の映像が 30 秒間流れる。
3. 真っ暗な映像が 20 秒間流れる。
4. 深海恐怖症の映像が 30 秒間流れる。
5. 真っ暗な映像が 20 秒間流れる。
6. 先端恐怖症の映像が 30 秒間流れる。
7. 真っ暗な映像が 20 秒間流れる。



图 10.2 高所恐怖症映像



图 10.3 深海恐怖症映像



图 10.4 先端恐怖症映像

(※文責: 松本柊哉)

### 10.3.3 解析プログラム

本プログラムは、ユーザがアプリケーション上で恐怖症に対応する映像を視聴した際に、スマートウォッチから得られた心拍数の時系列データを統計学的手法を用いて解析を行う。また解析結果からユーザが恐怖症映像に対して恐怖をどの程度感じるのかを推測する。プログラムの作成は Python を使用して行った。

プログラムの構成は3つに分けられる。1つ目は得られた心拍数の時系列データに移動平均フィルタを適用し、その結果を出力するプログラムである。2つ目はフィルタ適用後のデータに対して主成分分析を行い、その結果を出力するプログラムである。3つ目は主成分分析後のデータに対して、マハラノビス距離を用いたパターン認識を行い、認識結果を出力するプログラムである。

#### 解析の流れ

1. 学習・評価データの作成。
2. 学習・評価データに移動平均フィルタを適用。
3. フィルタを適用したデータを主成分分析で4次元に圧縮。
4. 圧縮したデータを用いてマハラノビス距離によるパターン認識を行う。
5. 認識結果を csv, txt ファイルに出力。

(※文責: 高杉拓海)

## 10.4 使用した技術

### 10.4.1 映像技術

360度カメラである RICOH THETA で撮影した高所の映像をアプリケーションを通して全天球型の映像として PC に保存し、球に映像を張り付けて動画をセットして再生した。

(※文責: 品川誠剛)

### 10.4.2 3DCG 技術

深海恐怖症、先端恐怖症の映像は、3DCG にて再現を行った。3D モデルの作成に使用した 3DCG 制作ツールと、映像内に登場する 3D モデルの入手元は以下に示す通りである。

- Blender[12]
  - Blender ウェブサイト にて入手可能
- アカシュモクザメ Scalloped Hammerhead Shark, S. lewini[13]
  - Skechefab にて入手可能

Blender とはオランダに本拠地を置く非営利団体である Blender Foundation が開発しているオープンソースの 3DCG 制作ソフトウェアである。Blender では、主に深海恐怖症映像の部屋、水中の柱、先端恐怖症映像の部屋、針のオブジェクトの作成を行った。また、深海恐怖症映像に使用したアカシュモクザメのモデルデータは、CC BY 4.0[18] で提供されている。オリジナルのアカ

シュモクザメのオブジェクトは、頂点数が膨大であった。そのため、このオリジナルのオブジェクトの改変を行った。具体的には、Blenderにて頂点数の削減を行い、その後FBX形式にてエクスポートし、Unityへインポートした。ポリゴンの削減手順については以下に示す。

1. 3DモデルデータをBlenderにインポートする。
2. オブジェクトが複数に分かれているため、一つに結合する。
3. オブジェクトを選択後、編集モードに遷移し、すべてのポリゴンを選択後、重複した頂点をマージする。
4. オブジェクトモードに遷移し、デシメートモディファイアーを追加する。その後、比率の項目を調整し頂点を削減する。
5. オブジェクトを選択後、編集モードにし、すべてのポリゴンを選択後、孤立を削除する。
6. 3DモデルをFBX形式でエクスポートする。

(※文責: 松本柊哉)

### 10.4.3 解析技術

各恐怖症に対応した映像をVRゴーグルで視聴した際に、スマートウォッチで視聴者の心拍数を計測し、得られたデータに対して前処理、主成分分析、マハラノビス距離を用いたパターン認識を行い視聴者の映像に対する恐怖の度合いを推測した。Pythonを用いて移動平均フィルタ、主成分分析、パターン認識を行った結果をファイルに出力するプログラムを作成した。以下に解析の手順、結果について記述する。

(※文責: 高杉拓海)

#### データの取得

プロジェクトメンバーの7人の20歳、21歳の男性を対象とした。被験者の計測前の平均心拍数を導出するため、VRゴーグルを装着したあと、真っ暗な画面を30秒間表示させた。30秒間の計測で得られた心拍数の時系列データから平均の心拍数を導出した。次に被験者に三つの恐怖症(高所、深海、先端)に対応した映像をVRゴーグルで視聴してもらった。一つの映像につき視聴中の30秒間と視聴後の20秒間の計50秒間の心拍数を計測した。心拍数データは1秒ごとに抽出した。抽出した一秒ごとの心拍数データから被験者の平均心拍数を引き、偏差を導出した。取得したデータはcsvファイルに出力した。

(※文責: 高杉拓海)

#### 学習データの作成

計測終了後、被験者に各映像にどれだけ恐怖を感じたかを3つの恐怖度で評価してもらった。恐怖度は映像に対して恐怖を深く感じた場合にHigh、恐怖を感じた場合にMid、恐怖を感じなかった場合にLowとした。評価結果から、各心拍数の時系列データのクラス分けを行った。クラス分けを行った21個の心拍数の時系列データを学習データとした。

(※文責: 高杉拓海)

## データの前処理

作成した学習データを平滑化するために移動平均フィルタを適用した。移動平均フィルタは Python の numpy ライブラリの `numpy.convolve` を用いて行った [17].

(※文責: 高杉拓海)

## 主成分分析

前期も同様に心拍数の時系列データを用いてパターン認識を行った。パターン認識を行った際に行列のランク落ちが発生し、解析を行うことが困難になった。そのため、ランク落ちを防ぐためにフィルタを適用した学習データを主成分分析を用いて、データの圧縮を行った。主成分分析のアルゴリズムについては第 4 章 3.1 節を参照。

(※文責: 高杉拓海)

## パターン認識・解析結果

視聴者の心拍数の時系列データから、映像に対してどの程度恐怖を感じているかを推測するためにマハラノビス距離によるパターン認識を用いて解析を行った。学習データを主成分分析で 4 次元に圧縮したものをを用いてパターン認識を行った。パターン認識のアルゴリズムについては第 4 章 3.3 節を参照。認識結果を以下に示す。

```
/----- count cla -----/  
6. 2. 0.  
5. 4. 1.  
0. 1. 2.  
  
/----- rate cla -----/  
0.28571429 0.0952381 0.  
0.23809524 0.19047619 0.04761905  
0. 0.04761905 0.0952381  
  
/----- 誤判別割合 -----/  
0.42857142857142855  
  
/----- 誤判別回数 -----/  
9
```

- count cla: 各クラスのデータがパターン認識後にどのクラスにどれだけ分類されたかを示す。
- rate cla: 認識後のクラスがデータ全体に占める割合を示す。
- 誤判別割合: 元のクラスとパターン認識後のクラスが異なっているデータの割合
- 誤判別回数: 元のクラスとパターン認識後のクラスが異なっているデータの個数

パターン認識を行った結果、誤判別回数は 9 回、誤判別割合は約 0.429 という結果が得られた。各クラスのデータで誤判別が起きており、誤判別割合も大きくなった。得られた結果から判別精度が悪いことがわかり、パターン認識の精度向上が課題となった。そこで精度向上のために学習デー

## X-Reality and Smartwatch Connecting Your Feelings and the World

タを 2 次元, 8 次元, 16 次元に圧縮し, 再度パターン認識を行った. 以下にそれぞれの解析結果を示す.

学習データを 2 次元に圧縮した際の結果

```
/----- count cla -----/
5. 2. 1.
2. 7. 1.
0. 1. 2.
/----- rate cla -----/
0.23809524 0.0952381 0.04761905
0.0952381 0.33333333 0.04761905
0. 0.04761905 0.0952381
/----- 誤判別割合 -----/
0.3333333333333333
/----- 誤判別回数 -----/
7
```

学習データを 8 次元に圧縮した際の結果

```
/----- count_cla -----/
7. 1. 0.
1. 9. 0.
1. 0. 2.
/----- rate_cla -----/
0.33333333 0.04761905 0.
0.04761905 0.42857143 0.
0.04761905 0. 0.0952381
/----- 誤判別割合 -----/
0.14285714285714285
/----- 誤判別回数 -----/
3
```

学習データを 16 次元に圧縮した際の結果

```
/----- count_cla -----/
8. 0. 0.
1. 9. 0.
0. 0. 3.
/----- rate_cla -----/
0.38095238 0. 0.
0.04761905 0.42857143 0.
0. 0. 0.14285714
/----- 誤判別割合 -----/
0.047619047619047616
```

/————— 誤判別回数 —————/

1

2次元に圧縮した際の誤り率は0.333となり、4次元に圧縮した時よりも誤り率が小さくなった。8次元に圧縮した際の誤り率は約0.143となり、さらに小さくなった。学習データを16次元に圧縮した際のパターン認識の誤り率が約0.048となり、各条件の中で最も小さいことが明らかになった。16次元に圧縮した場合、4次元に圧縮した場合よりも高精度の推測ができる可能性がある。次に計測した21個のデータから1個のデータを評価データとし、その他の20個のデータを学習データとしてパターン認識を行った。21個のデータそれぞれを評価データとした際の誤り率の平均は約0.714となった。

学習データの第一主成分と第二主成分を散布図で可視化したものと恐怖症映像視聴時の心拍数の時系列データをグラフにしたものを以下に示す。

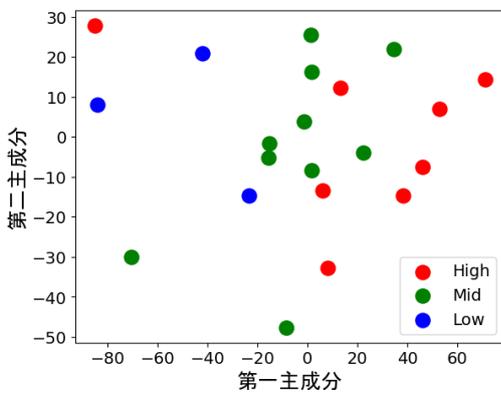


図 10.5 学習データの散布図

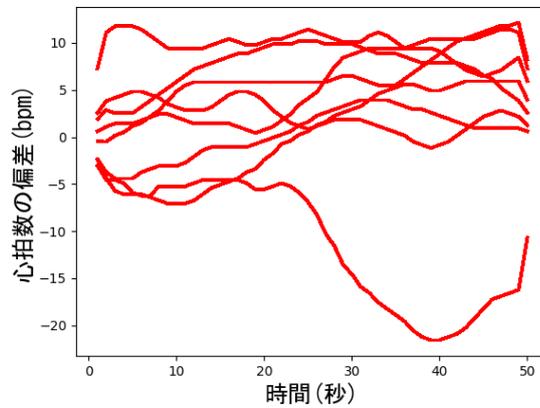


図 10.6 恐怖を深く感じた時の心拍数データ

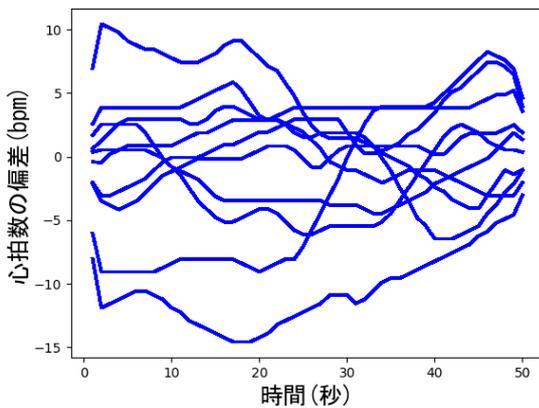


図 10.7 恐怖を感じた時の心拍数データ

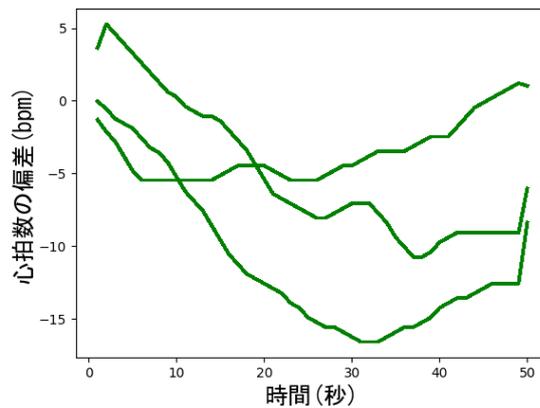


図 10.8 恐怖を感じなかった時の心拍数データ

図 10.5 の X 軸が第一主成分、Y 軸が第二主成分であり、High のデータが恐怖症映像に恐怖を深く感じた時のクラス、Mid のデータが恐怖を感じた時のクラス、Low のデータが恐怖を感じた時のクラスである。図からも心拍数から恐怖症映像に対する恐怖の度合いを推測することができる可能性があることが分かる。

図 10.6, 10.7, 10.8 の各グラフの X 軸は計測時間、Y 軸は計測した心拍数データからユーザーの平均心拍数を引いた偏差である。用いた偏差データには移動平均フィルタが適用されている。恐怖を深く感じた時の心拍数データのグラフからは、全体的に偏差が大きく、かつ正の方向にあり、上

昇傾向にあることがわかる。恐怖を感じた時の心拍数データからは、心拍数の上昇と下降を繰り返す傾向があることがわかる。恐怖を感じなかった時の心拍数データからは、全体的に偏差が負の方向にあり、下降傾向にあることがわかる。

(※文責: 高杉拓海)

### 考察と今後の展望

主成分分析で圧縮する次元数を変えて、パターン認識を行った結果、16次元に圧縮した際の誤り率が最も小さく、より高精度に推測ができると考えられた。しかし、評価データを用いてパターン認識を行った際に誤り率が約0.714と非常に大きくなった。誤り率が大きくなってしまった理由として、学習データの不足が影響していると考えられる。データ数が21個と少なく学習に十分な量とは言えない。加えて、3つのクラスで分別したが、クラスによってデータの量に差が出てしまった。このことから誤り率が高い原因はデータの不足にあると考えた。そのため、学習データのデータ数をさらに増やしていくべきである。また、恐怖症映像を見たときの心拍数データとそれ以外の映像を見たときの心拍数データの比較を行えず、計測したデータの有意性を示すことができなかった。認識結果の信頼性を担保するために、データの比較を行っていくべきである。計測方法についても今後検討していくべきである。

今回の計測、解析方法では、恐怖症映像を見た視聴者の映像に対する恐怖度の推測だけに留まった。本成果物の目標である、ユーザが恐怖症であるか否かの診断の実現はできなかった。恐怖症診断を実現するためには、恐怖症である人と恐怖症でない人の恐怖症映像視聴時の心拍数データを比較し、有意な差があるかを検証することが必要であると考えた。また、心拍数データ以外の身体データを取得し、それぞれのデータが恐怖の情動とどのような関係があるのかを精査していくことが必要であると考えた。

(※文責: 高杉拓海)

## 10.5 アプリケーション仕様書

### 10.5.1 概要

本アプリケーションはVRゴーグルとスマートウォッチを装着し、恐怖症に対応する映像を視聴したときの心拍数から、恐怖症を持っているか診断するアプリケーションである。

1. 心拍数受信・書き出しシステム
2. アプリケーション

心拍数受信・書き出しシステムのセットアップは第3章を参照。本アプリケーションは、上記の2つのシステムから構成されている。これらのシステムの開発環境は以下表10.1の通りである。

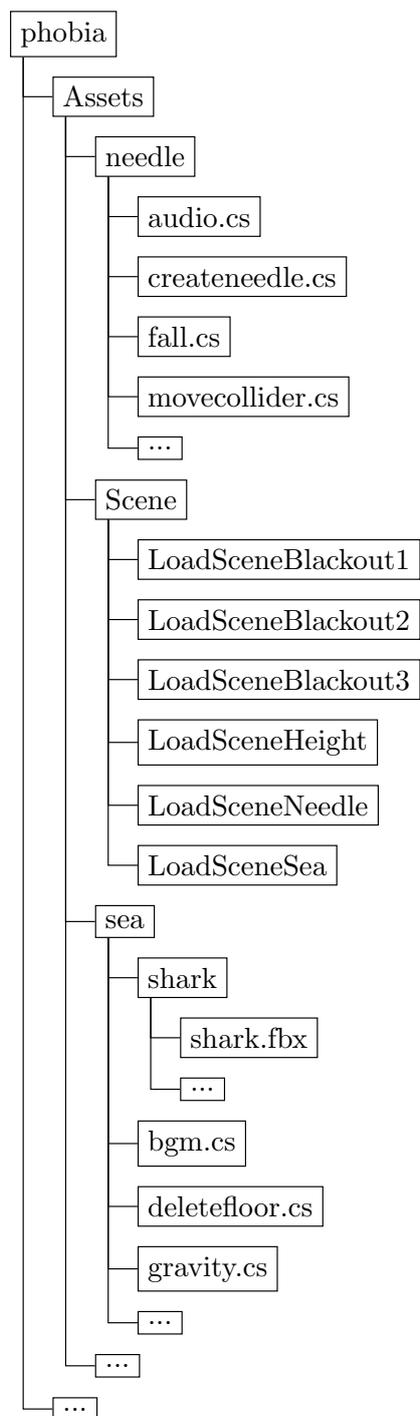
(※文責: 松本柊哉)

表 10.1 心拍数受信・書き出しシステム, アプリケーションの主な開発環境

仕様	心拍数受信・書き出しシステム	アプリケーション
OS	Windows10 Home	Windows11 Home
RAM	8GB	8GB
開発プラットフォーム	Java Script	Unity 2021.3.3f
node.js バージョン	16.15.1	-

## 10.5.2 ディレクトリ構造

心拍数受信・書き出しシステム及びアプリケーションのビルド済みファイルのディレクトリ構造は以下の通りである。なお、システムファイルやライブラリなどは記載を省略している。



(※文責: 松本柊哉)

### 10.5.3 プログラム詳細

10.5.2 節で記述した、ディレクトリ内にある各スクリプトの詳細を以下に示す。

**audio.cs** このファイルは、先端恐怖症映像において針オブジェクトが床に落ちた瞬間に、金属音を再生するスクリプトである。すべての針オブジェクトにアタッチすると音が大きすぎるため、あらかじめシーンに配置した7本の針オブジェクトにアタッチしている。

**createneedle.cs** このファイルは、先端恐怖症映像において、任意の二つのオブジェクトの x 軸と z 軸の座標を取り出し、その座標内のランダムな座標に針オブジェクトのプレハブを配置

するスクリプトである。

**fall.cs** このファイルは、先端恐怖症映像において、針オブジェクトが任意のコライダーにあたった際に、針オブジェクトにアタッチされているリジッドボディのグラビティをオンにし、落下させるスクリプトである。

**movecollider.cs** このファイルは、先端恐怖症映像内において、対象である針オブジェクトを落下させるためのコライダーを動かすスクリプトである。先端恐怖症映像は、頭上に針オブジェクトが設置され、一定時間が立つと奥側から順に落下してくるという流れである。そのためボックスコライダーをアタッチしたオブジェクトにこのスクリプトをアタッチし、奥から手前に移動させることで、針オブジェクトが落ちる仕組みになっている。

**LoadSceneBlackout1.cs** このファイルは高所恐怖症映像が 30 秒間再生された後の変化を計測するための、真っ暗な映像が流れるシーンに遷移するプログラムである。

**LoadSceneBlackout2.cs** LoadSceneBlackout1 と同様、深海恐怖症映像が再生された後にシーン遷移するためのプログラムである。

**LoadSceneBlackout3.cs** LoadSceneBlackout1 と同様、先端恐怖症映像が再生された後にシーン遷移するためのプログラムである。

**LoadSceneHeight.cs** このファイルは、アプリケーション起動時に、30 秒間平常時の心拍を計測するためのシーンから高所恐怖症映像のシーンへ遷移するためのプログラムである。

**LoadSceneSea.cs** このファイルは高所恐怖症映像による心拍の変化を、20 秒間計測した後に深海恐怖症映像のシーンへ遷移するためのプログラムである。

**LoadSceneNeedle.cs** LoadSceneSea と同様、先端恐怖症映像のシーンへ移行するためのプログラムである。

**bgm.cs** このファイルは、深海恐怖症映像において、水中での音を再生し、ループさせるスクリプトである。またプレイヤーが水中に落ちた際の水音も再生する。

**deletefloor.cs** このファイルは、深海恐怖症映像が再生されてから五秒後にプレイヤーが立っている足場を消すスクリプトである。これによってプレイヤーは水中へ落下する。

**gravity.cs** このファイルは、深海恐怖症映像において、deletefloor.cs のスクリプトで足場が消えた後水中へ落下した後、重力の大きさを変更し、水中のような浮力を再現するためのスクリプトである。

(※文責: 松本柊哉)

## 10.6 解析プログラム仕様書

### 10.6.1 概要

本プログラムは、アプリケーションから得られた心拍数の時系列データから、統計学的手法を用いて解析を行い、ユーザが映像に対してどの程度の恐怖を感じているかを推測するプログラムである。

(※文責: 高杉拓海)

## 10.6.2 解析プログラムの開発環境

本プログラムの開発環境については表 10.2 の通りである。

表 10.2 解析プログラムの主な開発環境

仕様	解析プログラム
OS	Windows11 Home 64bit
RAM	8GB
開発プラットフォーム	Python 3.11.6

(※文責: 高杉拓海)

## 10.6.3 プログラムの詳細

本プログラムでは以下の三つのプログラムで構成されている。

**MA\_filter.py** データに移動平均フィルタを適用するプログラム。

**pca.py** 主成分分析を行うプログラム。

**RecogLDF.py** マハラノビス距離によるパターン認識を行うプログラム。

以下に各プログラムの詳細について示す。

### MA\_filter.py

**data\_read 関数** pandas ライブラリを使用し、csv ファイルの読み込みを行う。csv ファイルの心拍数の時系列データ部分の抽出、抽出したデータの数値化を行う。

**MA\_filter 関数** data\_read 関数から返されたデータに対して移動平均フィルタを適用する。numpy ライブラリの convolve 関数を用いて移動平均フィルタを実装している。

**MA\_out** MA\_filter 関数から返された移動平均を適用した結果と各データのクラスを csv ファイルに出力する。

### pca.py

**file 関数** pandas ライブラリを使用し、csv ファイルの読み込みを行う。

**data\_read 関数** file 関数から返されたデータの心拍数の時系列データ部分を抽出し、抽出したデータの数値化を行う。

**cov 関数** data\_read 関数から返された配列の共分散行列を計算する。numpy ライブラリの cov 関数を用いて導出している。

**eigen 関数** cov 関数から返される共分散行列の固有値と主軸行列を導出する。固有値は大きい順にソートされ、主軸行列も固有値の大きい順にソートされる。導出には numpy ライブラリの linalg.eigh という関数を用いている。

**pca 関数** eigen 関数から返された主軸行列と data\_read 関数から返される行列の積を求め、主成分ベクトルを導出する。抽出する主成分ベクトルの数を指定して、返すことができる。

**pca\_out 関数** pca 関数で導出された主成分ベクトルと各データのクラスを csv ファイルに出力する。

### RecogLDF.py

**data\_read 関数** pandas ライブラリを用いて csv ファイルの読み込みを行う。

**ext\_pred 関数** 読み込んだ csv ファイルの主成分ベクトル部分の抽出を行い、数値化をする。

**make\_list 関数** 読み込んだデータの各クラスを抽出し、抽出したクラスのリストを返す。

**ext\_L 関数** 学習データから共通クラスを持つデータを抽出する。主成分ベクトルの部分とクラス部分に分け、主成分ベクトル部分を返す。

**calc\_sl 関数** ext\_L から返された行列の共分散行列を導出する。

**calc\_ml 関数** 各クラスの重心 (標本平均ベクトル) を導出する。ext\_L から返された行列の列方向の平均値を導出する。重心の導出については、第 4 章 3.3 節の式 4.29 を参照。

**calc\_SL 関数** 標本分散共分散行列を導出する。標本分散共分散行列の導出については、第 4 章 3.3 節の式 4.31 を参照。

**calc\_Mdistance 関数** 各クラスの重心から各データまでのマハラノビス距離を導出する。マハラノビス距離の導出については、第 4 章 3.3 節の式 4.32 を参照。

**decide\_class 関数** 各データを calc\_Mdistance 関数から返されたマハラノビス距離が最も小さいクラスに分別する。判別後のクラス、正しい判別だった場合に TRUE、誤判別だった場合に FALSE を配列に格納する。判別したクラスと元のクラスが異なっていた回数と割合を導出する。各クラスのデータがどのクラスにいくつ判別されたかを記録した行列と、全データ数に対する割合を求めた行列を導出する。

**recogldf\_output** decide\_class 関数から返される結果を txt ファイルと csv ファイルに出力する。

(※文責: 高杉拓海)

## 10.7 まとめ

B 班では、本プロジェクトの目的である「X-Reality とスマートウォッチを用いて身の回りの現象に接続し、自分の気持ちと世界をつなぐための情報リンクを作成する」ことに沿った成果物の制作を行うことを目的とした。そのために、それぞれの技術を統合した成果物である「恐怖症診断」を制作を目指した。VR ゴーグルに投影された恐怖症に対応する映像を見たときの心拍数データを用いて解析を行い、ユーザが映像にどの程度の恐怖を感じたかを推測するシステムの開発を行った。

解析結果から、恐怖症映像を視聴した際の心拍数データから、ユーザが映像に対してどの程度の恐怖を感じたかの推測はできると考えられた。しかし、恐怖度の推測だけに留まってしまい、当初の目標である恐怖症診断は実現できなかった。また、学習データの有意性を示すことができていない、データ数が不十分であるなどの問題があるため、恐怖症診断実現のためにこれらの問題を解決していく必要がある。

この章では、B 班の活動内容のまとめや各メンバーの反省を述べる。

(※文責: 品川誠剛)

### 10.7.1 映像担当のまとめ

映像班では、Unity を用いて恐怖症診断に必要な恐怖症映像の制作や、映像の切り替え等のシステム面に取り組んだ。反省点としては、恐怖症映像を3つ用意したが、幅広く診断するために恐怖症映像の数を幾つか増やすべきだった。また、映像の中で恐怖を感じさせる工夫をもう少し行うべきだった。

(※文責: 品川誠剛)

### 10.7.2 スマートウォッチ担当のまとめ

スマートウォッチ班では、スマートウォッチで得られる心拍数データの取得と記録に取り組んだ。結果としては、心拍数の計測とデータの出力ができたため、スマートウォッチ班の役割はおおむね果たすことができた。

(※文責: 松本柗哉)

### 10.7.3 解析担当のまとめ

解析班では、映像を聴時した際の心拍数の時系列データの計測、解析に取り組んだ。心拍数の時系列データから恐怖症映像に対する恐怖度の推測ができた。しかし、当初の目的であった恐怖症の診断は実現するためには、データの計測方法、解析手法について検討していく必要がある。また、計測したデータの有意性を示すことができなかつたり、データの収集が不十分であつたり、課題が残る結果になってしまった。反省点としては、計測方法の見直しが遅れ、作業の進行が遅れたため、データの計測方法について早い段階から教員とコミュニケーションをとりながら検討していくべきだった。

(※文責: 高杉拓海)

### 10.7.4 各メンバーの反省

環境構築に時間を取り過ぎてしまったので、時間外の作業を増やすべきだった。

パソコンや Unity の知識が浅く、班員の足を引っ張ってしまった。

(※文責: 品川誠剛)

映像制作のスケジュール管理が上手くいかず、当初目標としていた映像数よりも少なくなつてしまった。

(※文責: 松本柗哉)

教員とのコミュニケーションを積極的に行うことができず、自身の取り組みの見直しが遅れた結果、プロジェクトの進行を遅延させてしまった。

(※文責: 高杉拓海)

## 第 11 章 C 班

この章では、C 班の後期の活動内容について述べる。

(※文責: 川端勇雅)

### 11.1 目的・目標

C 班では、アプリケーションを開発し、ユーザーの睡眠の促進・目覚めの改善・睡眠の見直しを目的とする。その目的を達成するために、前期の三つのグループ活動を生かし、スマートウォッチや活動量計からリアルタイムで取得した心拍数を用いたアプリケーション開発を目標とする。

1. 心拍数を扱う開発
2. Unity を利用した開発
3. 心拍数の解析

上記の三つがアプリケーション開発の具体的な内容であり、「1. 心拍数を扱う開発」は前期の SW 班、「2. Unity を利用した開発」は前期の映像班、「3. 心拍数の解析」は前期の解析班が担当した。

(※文責: 川端勇雅)

### 11.2 テーマの決定

前期までの活動をふまえて、C 班はアプリケーションのテーマを決定した。まずアプリケーションのアイデアを出し合った。そこで出たアイデアの抜粋を下記に示す。

- 消費カロリーや心拍数データをもとに運動をサポートするアプリケーション
- 消費カロリーに応じて敵を倒すアプリケーション
- 心拍数データによって睡眠を促進するアプリケーション

これらのアイデアの中で、心拍数データによって睡眠を補助するアプリケーションに着目した。睡眠時にはレム睡眠とノンレム睡眠という睡眠の深さによって判別できる二つの状態がある。この二つの状態を判別する要因として心拍数を示唆している論文がある [14]。そこで我々は心拍数のリアルタイムの計測によりレム睡眠とノンレム睡眠の判別を試みた。このアイデアは前期で習得した内容で比較的簡単に実装することができ、効率的に成果を出せると判断したため、心拍数データによって睡眠を補助するアプリケーションを開発することに決定した。

(※文責: 川端勇雅)

## 11.3 成果物

### 11.3.1 アプリケーション

本アプリケーションは、スマートウォッチをつけながら行うアプリであり、スマートウォッチと VR の特性を生かした睡眠の補助を行うことを目的にしている。具体的な睡眠の補助の内容は下記に記載する。

(※文責: 長澤颯音)

#### 睡眠の促進

睡眠の促進を行うために、睡眠時のシーンで睡眠を促進させる背景と BGM を表示する予定であった。今回は時間不足により、促進させる効果が証明されていない背景を仮で表示させた。その際に、背景素材である、AllSky Free - 10 Sky / Skybox Set[9] を使用した。

(※文責: 長澤颯音)

#### 目覚めの改善

レム睡眠時に目覚めると、目覚めがよくなるといわれている。そこで、レム睡眠時にアラームを鳴らすことにより目覚めの改善を試みた。レム睡眠時の判別方法は、心拍数を用いて行った。レム睡眠時は睡眠中の心拍数が比較的多くなるため、アプリの最初に心拍数の設定を行ってもらい、睡眠中に心拍のデータをリアルタイムで取得し、設定した心拍数以上になるとアラームが鳴るようにした。そのためアプリの利用には、レム睡眠時の平均心拍数を知っておく必要がある。

(※文責: 長澤颯音)

#### 睡眠の見直し

アプリの実行を一通り行うと、睡眠中の心拍数を 10 分ごとに記録した CSV ファイルが出力される。その CSV ファイルを解析して睡眠が浅い傾向にあるか、深い傾向にあるかの診断を行う。具体的な方法は、11.4.3 にて記載する。

(※文責: 長澤颯音)

### 11.3.2 成果物の全体の流れ

このセクションでは、C 班が作成した成果物を利用するための一連の流れを記載する。

(※文責: 長澤颯音)

#### アプリ起動までの流れ

1. 付録（心拍を測る手順の場所を参照）に従い、心拍数が取得できている状態を想定する。
2. 使用者は任意のスマートウォッチ（ANT+ 規格の通信を用いて心拍数を送信できるもの）

## X-Reality and Smartwatch Connecting Your Feelings and the World

を用意し、装着する。本プロジェクトでは garmin 社製のスマートウォッチ vivosmart5 を使用した。

3. ゲームを実行するパソコンにドングルを装着する。
4. スマートウォッチの設定を心拍数が送信できる設定に変更する。
5. 心拍数受信システム (mainjs.bat) を起動し、心拍数が正しく受信できていることを確認する。
6. アプリを起動する。

(※文責: 長澤颯音)

### アプリケーションの流れ

1. 自分のレム睡眠時の平均心拍数を知っていることを前提とする。
2. ゲームを起動すると心拍数とアラームを設定するシーン (図 11.1) が表示される。このシーンは以下、設定シーンと表記する。このシーンでは、レム睡眠時の平均心拍数と起きたい時刻を設定して、「Button」を押すことによって、次のシーン (図 11.2) へ遷移する。このシーンは以下、睡眠シーンと表記する。
3. 睡眠シーンでは、現在時刻と背景が表示される。設定したアラームの時刻以降に心拍数が設定したレム睡眠時の平均心拍数を超えると、アラーム音が鳴るようになっている。
4. アラームが作動すると同時にボタンも出現する。出現したボタンを押すと、アラーム音が止まると同時に睡眠時の心拍数を記録した CSV ファイルが出力される。

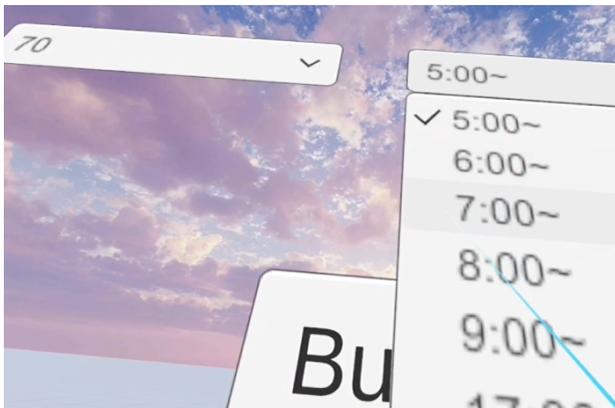


図 11.1 設定シーン



図 11.2 睡眠シーン

アプリケーションの流れを図にまとめたものを以下に示す。

(※文責: 長澤颯音)

### 解析の流れ

1. 学習データの作成を行う。詳細は 11 章 4.5 節、「学習データの作成」を参照。
2. 学習データに移動平均フィルタをかけ、主成分分析によって 4 次元に圧縮する。
3. 作成した学習データを用いてマハラノビス距離によるパターン認識でパターン認識を行う。
4. 解析結果のファイルが出力される。

## ツールの流れ

### 1, 起床時刻の設定

- アラームの設定
- レム睡眠時の心拍を設定

### 2, 睡眠・心拍数計測開始

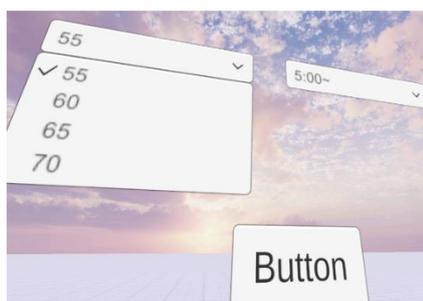
- スマートウォッチを装着

### 3, 睡眠促進のための音楽・映像放映

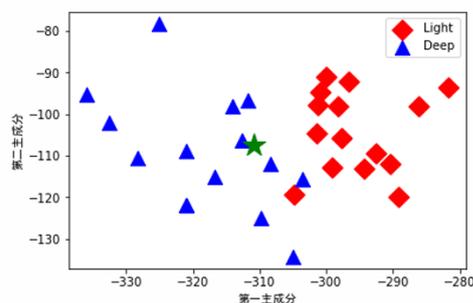
- 360度動画の放映(星空、自然など)

### 4, 起床・解析結果の表示

- 深い睡眠と浅い睡眠の判別
- 良い起床ができる  
レム睡眠時にアラーム
- パターン認識により  
睡眠の質を評価、表示



アプリ内、アラーム・心拍設定画面



深い睡眠・浅い睡眠のパターン認識の散布図

図 11.3 アプリケーションの流れ

(※文責: 丹下裕太)

## 11.4 使用した技術

### 11.4.1 Unity で使用した機能

本アプリケーションは、Unity を使用して作成した。ここではアプリの制作に使用した特徴的な機能について記載する。

(※文責: 長澤颯音)

### 11.4.2 DateTime 構造体

DateTime は C# で使われる構造体である。年月日、時、分、秒のデータを入れることができる。DateTime は変数なので時刻の比較を行うこともできる。DateTime.Now で現在の時刻を取得することもできる。

(※文責: 長澤颯音)

### 11.4.3 PlayerPrefs

PlayerPrefs は Unity 上で使われる、データを保存するための機能である。PlayerPrefs はキーと値のペアで構成されており、例えば、`PlayerPrefs.SetInt("A", 10);`と書くと、キーが A、値が Int 型の 10 の PlayerPrefs が生成される。参照や呼び出すときには、`GetInt("A")`と書くことにより 10 の値を得ることができる。この機能を用いることで、シーン間をまたいで参照することのできる変数を作り出すことができる。

(※文責: 長澤颯音)

### 11.4.4 ドロップダウン

ドロップダウンとは、選択肢を表示させてプレイヤーに選ばせるために用いられる UI の一部である。Unity の中で UI を配置するために必要な Canvas を用意し、その中にドロップダウンを配置する。選択肢の追加、削除、項目の変更は自由に行える。ドロップダウンの中には、選択が変更されたときに実行される `OnValueChanged` メソッドが用意されている。このメソッドは現在選択している選択肢を表す整数値が引数として使用されている。最初の選択肢が選ばれている場合は 1、次の選択肢の場合は 2、といったように続いている。この文章は、「Unity ユーザーマニュアル」内の Dropdown ページ [10] を参考にしている。

(※文責: 長澤颯音)

#### アプリケーションでの実装方法

設定シーンで心拍数とアラームを設定する画面が表示される。この設定を行う機能は、ドロップダウンを用いて実装した。心拍数を設定するドロップダウンとアラームを設定するドロップダウンの2つを作成した。心拍数を設定するドロップダウンには、55,60,65,70 のオプションを順に追加し、アラームを設定するドロップダウンには、5:00,6:00,7:00,8:00,9:00,10:00 のオプションを順に追加した。また、ボタンを押した際に `On Value Changed` イベントが発生し、PlayerPrefs にそれぞれのドロップダウンで選択したオプションのインデックスを表す整数値を代入するペアを二つ作成した。PlayerPrefs の変数に代入することにより、前のシーン（設定シーン）で設定した値を次のシーン（睡眠シーン）に引き継ぐことが可能になった。今回は、キーを"Setshinpaku"に指定した PlayerPrefs に心拍数のドロップダウンの `OnValueChanged` の引数を値に入れて、キーを"Settime"に指定した PlayerPrefs にアラームのドロップダウンの `OnValueChanged` の引数を値に入れた。睡眠シーンでは `Date Time.Now` を用いて現在時刻を表示した。アラームの仕組みは、キーが"Settime"の PlayerPrefs の値に応じた時刻と現在時刻が一回でも一致した時、かつキーが"Setshinpaku"の PlayerPrefs の値に応じた心拍数よりも現在の心拍数が高いときにアラームを鳴らすように設定した。

(※文責: 長澤颯音)

### 11.4.5 解析技術

睡眠促進アプリケーションを使用し、得られた心拍データをもとに、睡眠診断を行った。Python を用いて移動平均フィルタ、主成分分析、パターン認識とそれぞれの結果をファイルに出力するプログラムを作成した。

(※文責: 丹下裕太)

#### 学習データの作成

睡眠診断を行うにあたって、学習データの作成を行った。はじめに、被験者が睡眠を行い、アプリケーションとスマートウォッチを用いて睡眠時の心拍数を測定を行う。計測した心拍データを10分ごとに抽出する。次に、被験者が起床後、よく眠れたか、そうでないかを主観で判断する。これをもとに Deep と Light の二つにクラス分けを行う。

この作業を何度か繰り返し、学習データを作成した。また、十分な量のデータが確保できなかった。それを解消するために、データに作成者の主観に基づいて作成されたダミーデータが含まれている。そのため、データの信頼性には注意が必要である。

#### データの解析

マハラノビス距離によるパターン認識を用いて、睡眠診断を行った。パターン認識のアルゴリズムについては、第4章3.3節を参照。

また、データの前処理として、学習データに移動平均フィルタをかけ、さらに主成分分析を行ってデータを4次元まで圧縮している。主成分分析のアルゴリズムについては、第4章3.1節を参照。また、移動平均については、以下を参照。

#### 移動平均

作成した学習データを平滑化するために、データに移動平均フィルタを適用した。移動平均は、Python の Numpy にある、`numpy.convolve` と呼ばれる関数を用いて求めている [17]。

#### 解析結果

学習データ同士でパターン認識を行った結果を以下に示す。

```

/----- count cla -----/
16. 1.
2. 16.
/----- rate cla -----/
0.45714286 0.02857143
0.05714286 0.45714286
/----- 誤判別割合 -----/
0.08571428571428572
/----- 誤判別回数 -----/
3

```

次に、パターン認識を行った結果を散布図上に可視化したものを以下に示す。

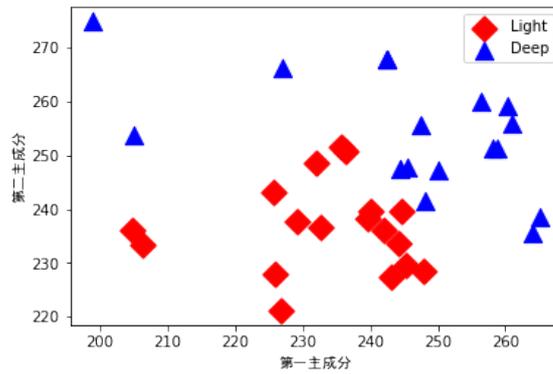


図 11.4 パターン認識結果

X 軸が第一主成分、Y 軸が第二主成分であり、赤色のデータが浅い睡眠のクラス、青色のデータが深い睡眠のクラスである。

図によると、浅い睡眠と深い睡眠が心拍数によって判別できる可能性があることがわかる。

以下に、睡眠時の心拍数の時系列データをグラフにしたものを示す。これらの図によると、心拍数の変化の様子は浅い睡眠と深い睡眠の時で同じような変化をしている可能性があることが分かる。

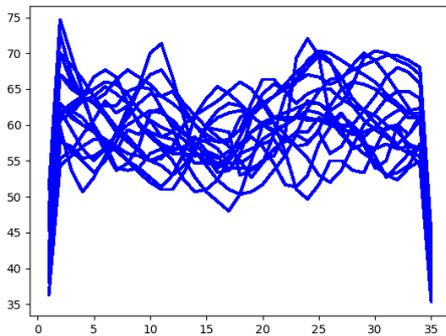


図 11.5 浅い睡眠の時の心拍数データ

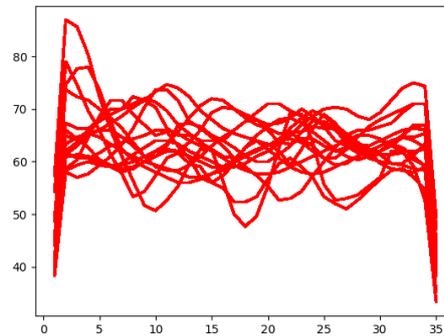


図 11.6 浅い睡眠の時の心拍数データ

(※文責: 丹下裕太)

## 11.5 仕様書

### 11.5.1 概要

本グループにおけるアプリケーションは、ユーザーの睡眠の促進・目覚めの改善・睡眠の見直しを目的とするアプリケーションである。本アプリケーションの名称は「睡眠促進ツール」である。本アプリケーションの開発環境は以下の表の通りである。

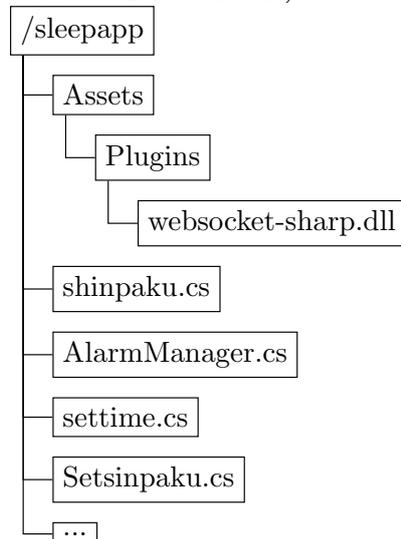
(※文責: 丹下裕太)

表 11.1 睡眠促進アプリケーションの主な開発環境

仕様	心拍数受信システム	アプリケーション	データ解析
OS	Windows 11 Home 64bit		
RAM	8GB 以上		
開発プラットフォーム	JavaScript	Unity 2022.3.1f1	Python 3.7.9
node.js バージョン	18.16.0	-	-

### 11.5.2 ディレクトリ構成

心拍数受信システム及び Unity プロジェクトファイルのディレクトリ構造は以下の通りである。なお、システムファイルやライブラリは記載を省略している。不手際により完成ファイルに不具合が起きてしまったため、csv が出力されない予備の Unity プロジェクトファイルを配置している。



(※文責: 長澤颯音)

### 11.5.3 ソースコード詳細

以下はアプリケーションで使用した shinpaku.cs のソースコードの一部である。特に、11.4.1,2,3,4 にかかわる部分を抜粋した。なお行切れを修正するために一部改変が行われている。

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using System;

public class AlarmManager : MonoBehaviour
{

```

```
public Text currentTimeText;
public Button stopAlarmButton;
public bool alarmflag = false;
public bool stop = false; ボタンが押されたら

//にするtrue
[SerializeField] private AudioSource audioSource;
public float setsinpaku = 100;

//PlayerScript playerscript; 呼ぶスクリプトにあだなつける

//
//GameObject obj = GameObject.Find("shinpaku");

//っていうオブジェクトを探すPlayer
//playerscript = obj.GetComponent<PlayerScript>(); スクリプトを取得

//

public Shinpaku Script;

// アラーム時刻を型に変換DateTime

DateTime alarmDateTime = DateTime.Parse("05:00:00");

//DateTime alarmDateTime = DateTime.Parse("17:29:00");

public int sinpa;
public int set;

void Awake()
{

set = PlayerPrefs.GetInt("Setshinpaku");
sinpa = PlayerPrefs.GetInt("Settime");

}

void Start()
```

```
{

alarmDateTime = DateTime.Parse("05:00:00") + TimeSpan.FromHours(set);

setsinpaku = sinpa*5 + 50 ;

}

void Update()
{

// 現在時刻を取得して表示
currentTimeText.text = DateTime.Now.ToString("HH:mm:ss");

// 現在の時刻を取得
DateTime currentTime = DateTime.Now;

// 現在時刻とアラーム時刻を比較時刻が設定時刻以降
if (currentTime.Hour == alarmDateTime.Hour &&
currentTime.Minute == alarmDateTime.Minute)
{

alarmflag = true;

}

if (stop == false &&
Script.average > setsinpaku && alarmflag == true)ストップボタン
が押されていないとき
{//

stopAlarmButton.gameObject.SetActive(true);ストップボタンを有効化

//

audioSource.Play();

}

}
```

```
public void stopAlarmボタンが押されたときに発動()//
{
    stop = true;
    alarmflag = false;
    audioSource.Stop();
}
}
```

(※文責: 長澤颯音)

#### 11.5.4 解析技術の仕様書

##### 解析のプログラムの詳細

本プログラムは、以下の三つのプログラムから構成されている。

##### MAFilter.py

データに移動平均フィルタを適用するプログラム。

##### pca.py

主成分分析を行うプログラム。

##### RecogLDF.py

マハラノビス距離によるパターン認識を行うプログラム。

以下にプログラムの詳細について示す。

##### MAFilter.py

**data\_read 関数** pandas ライブラリを使用し、csv ファイルの読み込みを行う。csv ファイルの心拍数の時系列データ部分の抽出、抽出したデータの数値化を行う。

**MA\_filter 関数** data\_read 関数から返されたデータに対して移動平均フィルタを適用する。numpy ライブラリの convolve 関数を用いて移動平均フィルタを実装している。

**MA\_out** MA\_filter 関数から返された移動平均を適用した結果と各データのクラスを csv ファイルに出力する。

##### pca.py

**file 関数** pandas ライブラリを使用し、csv ファイルの読み込みを行う。

**data\_read 関数** file 関数から返されたデータの心拍数の時系列データ部分を抽出し、抽出したデータの数値化を行う。

**cov 関数** data\_read 関数から返された配列の共分散行列を計算する。numpy ライブラリの cov 関数を用いて導出している。

**eigen 関数** cov 関数から返される共分散行列の固有値と主軸行列を導出する。固有値は大きい順にソートされ、主軸行列も固有値の大きい順にソートされる。導出には numpy ライブラリの `linalg.eigh` という関数を用いている。

**pca 関数** eigen 関数から返された主軸行列と `data_read` 関数から返される行列の積を求め、主成分ベクトルを導出する。抽出する主成分ベクトルの数を指定して、返すことができる。

**pca\_out 関数** `pca` 関数で導出された主成分ベクトルと各データのクラスを csv ファイルに出力する。

## RecogLDF.py

**data\_read 関数** pandas ライブラリを用いて csv ファイルの読み込みを行う。

**ext\_pred 関数** 読み込んだ csv ファイルの主成分ベクトル部分の抽出を行い、数値化をする。

**make\_list 関数** 読み込んだデータの各クラスを抽出し、抽出したクラスのリストを返す。

**ext\_L 関数** 学習データから共通クラスを持つデータを抽出する。主成分ベクトルの部分とクラス部分に分け、主成分ベクトル部分を返す。

**calc\_sl 関数** `ext_L` から返された行列の共分散行列を導出する。

**calc\_ml 関数** 各クラスの重心 (標本平均ベクトル) を導出する。`ext_L` から返された行列の列方向の平均値を導出する。重心の導出については、第 4 章 3.3 節の式 4.29 を参照。

**calc\_SL 関数** 標本分散共分散行列を導出する。標本分散共分散行列の導出については、第 4 章 3.3 節の式 4.31 を参照。

**calc\_Mdistance 関数** 各クラスの重心から各データまでのマハラノビス距離を導出する。マハラノビス距離の導出については、第 4 章 3.3 節の式 4.32 を参照。

**decide\_class 関数** 各データを `calc_Mdistance` 関数から返されたマハラノビス距離が最も小さいクラスに分別する。判別後のクラス、正しい判別だった場合に TRUE、誤判別だった場合に FALSE を配列に格納する。判別したクラスと元のクラスが異なっていた回数と割合を導出する。各クラスのデータがどのクラスにいくつ判別されたかを記録した行列と、全データ数に対する割合を求めた行列を導出する。

**recogldf\_output** `decide_class` 関数から返される結果を txt ファイルと csv ファイルに出力する。

(※文責: 丹下裕太)

## 11.6 まとめ

この章では、C 班の活動のまとめと各メンバーの反省を述べる。

C 班では、本プロジェクトの目的である、X-Reality とスマートウォッチを用いて身の回りの減少に接続し、自分の気持ちと世界をつなぐための情報リンクを作成することに沿った成果物の制作を行うことを目的とした。そのために、各メンバーの技術を統合し、成果物である睡眠促進ツールを作成した。

結果として、心拍数を用いて睡眠を可視化することができたが、十分なデータ数が確保できなかったため、結果の信頼性には議論が必要である。

また、当初の目的である睡眠の促進、目覚めの改善、睡眠の見直しについて、達成できたかの評価を行うまでに至らなかった。

(※文責: 丹下裕太)

### 11.6.1 映像担当のまとめ

睡眠時の心拍数を参照してアラームを鳴らすアプリを Unity で作成した。睡眠を促進させる背景は検証に時間が取れなかったため、仮の背景を表示させた。

(※文責: 長澤颯音)

### 11.6.2 スマートウォッチ担当のまとめ

睡眠時の心拍数のデータをリアルタイムで利用するため、スマートウォッチから Unity へ心拍のデータをリアルタイムで送信できるようにした。

(※文責: 川端勇雅)

### 11.6.3 解析担当のまとめ

心拍数を用いた深い睡眠と浅い睡眠の区別について、学習データ同士でパターン認識を行った結果、誤り率は 0.086 だった。つまり、心拍数で浅い、深い睡眠が区別できている可能性があるということがわかる。しかし、データ中にダミーデータが含まれている、データ数が不足している点に注意する必要がある。

また、様々な角度からデータを集め、今回のデータの有意性の比較を行うことができなかった。

(※文責: 丹下裕太)

### 11.6.4 各メンバの反省

心拍数を参照する最低限の機能は完成させたが、睡眠促進の根拠のある背景や、レム睡眠時の心拍数の自動計測などの実装に至ることができなかった。

(※文責: 長澤颯音)

Unity へ心拍数を送信することができたが、SW 班として成果物に貢献できる点が少なかった。

(※文責: 川端勇雅)

Unity 上から解析を行いたかったが、技術と時間が足りず、そこまで実装に至らなかった。また、アプリケーションそのものの作成に貢献できなかった。

(※文責: 丹下裕太)

## 第 12 章 成果発表会

この章では後期に行った（2023，12，8）成果発表会について述べる。成果発表会は、これまでのプロジェクト学習の成果を学内外の関係者に発表し、評価していただくことを目的としたものである。本プロジェクトは主にスライドとポスターを用いて発表し、また発表会終了後、評価者に記入していただいた評価フォームをもとに、集計・解析・検討を行った。

（※文責: 川端勇雅）

### 12.1 成果発表会に向けて

まず、A 班と B 班では実際に SW を利用してアプリを動作させるため、本番で動作しない場合も考えられる。それに対策し、あらかじめ動作しなかった場合の動画を用意した。C 班ではデモは事前に準備した動画を使用した。加えて、SW を利用する際、複数台を利用すると混線して正しい値を示さなくなるため、三班が同時に発表するのは困難である。したがって、三班がスムーズに交代して発表するために、プロジェクターへの接続の練習を行った。

（※文責: 川端勇雅）

### 12.2 ポスター・スライドの作成

成果発表会では、A 班・B 班・C 班それぞれ一枚ずつのサブポスターと、全体のメインポスター一枚を作成した。この 4 枚のポスターは Word で作成した。また、成果発表会で使用したスライドは PowerPoint で作成した。

（※文責: 川端勇雅）

### 12.3 成果発表会の発表形式

本プロジェクトでは、前半 5 人（新田，川端，品川，丹下，米森）と、後半 4 人（結城，長澤，高杉，松本）に分かれ、15 分間で発表・質疑応答を 3 回ずつ行った。

（※文責: 川端勇雅）

### 12.4 フィードバック

前後半合わせて 43 枚の評価フォームが集まった。評価者の内訳は、本学の学生が 35 枚、教員が 6 枚、一般の方が 2 枚であった。集まった評価フォームから発表技術の評価とプロジェクトの評価について分析、考察を行った。

### 12.4.1 発表技術のフィードバック

発表技術について評価点（10点満点）は平均値が7.23（有効数字3桁）、中央値が7であった。コメントのうち高評価であったもの（8点以上）の中で一部抜粋したものを以下にまとめる。

- デモがわかりやすかった。
- 成果物の内容が簡潔でよく分かる発表だった。
- 動画資料を効果的に用いていた。
- 解析の流れがわかりやすかった。
- デモを観客ができると思った。
- スライドが見やすかった。
- どのようなプロセスで計測を行うのかや、本当に必要な情報のみを提供していたので、よかったと感じた

次にコメントのうち低評価であったもの（6点以下）の中で一部抜粋したものを以下にまとめる。

- 時間配分が適切でなかった。
- 声が小さく聞き取りづらかった。
- 専門用語や難しい言葉が多くて分からないことがあった。
- どのデータが何を示しているのか分かりにくい。

### 12.4.2 プロジェクトのフィードバック

プロジェクトについて評価点（10点満点）は平均値が7.95（有効数字3桁）、中央値が8であった。コメントのうち高評価であったもの（8点以上）の中で一部抜粋したものを以下にまとめる。

- 実際に見て学ぶことのできる発表であった。
- 成果物を使用するだけでなく、そこで得られたデータから考察も行って素晴らしいと思った。
- 仮想空間は本人のヘルスデータと結びついていいため、この取り組みは意義のあるものだった。
- コンセプトや大学数学をフル活用した計算には関心させられた。
- 使用技術に関しての説明がしっかりされていた。
- 斬新なアイデアで面白いと思った。
- 目的とそれに対するアプローチが良かったと思う。
- テーマに基づいた成果物が作られているのがわかった

次にコメントのうち低評価であったもの（6点以下）の中で一部抜粋したものを以下にまとめる。

- スマートウォッチからとれるデータが心拍数しかないのが、少し物足りなく感じた。

- 可視化して何をおこないたいのかがわからない。
- 検討中となっている項目がいくつか見られた。

(※文責: 米森麻翔)

### 12.4.3 フィードバックを受けて

上記例をはじめとしたフィードバックを受けた結果についての解析、評価について以下に示す。

(※文責: 米森麻翔)

#### 発表技術の評価

発表全体を通して、「デモを用いていたためわかりやすかった」、「スライドが見やすかった」など視覚的な面でポジティブなコメントを多くいただいた。中間発表でも同じくスライドが見やすいとのコメントをいただいていたため、成果発表ではより見やすく、わかりやすいビジュアルを意識してきた。結果、成果発表会でも見やすかったというコメントを多くいただくことができた。このことから聴衆が一目見ただけで理解しやすい発表を行うことができただろう。しかし、本プロジェクトの発表場所が体育館という複数のプロジェクトが同時に発表する場であったこともあり「声が小さくて聞きづらかった」という指摘を多くいただいた。声の大きさに関する指摘は中間発表のときから多くいただいていたため、改善策を模索すべきであった。

(※文責: 米森麻翔)

#### プロジェクトの評価

プロジェクトの内容について、「斬新なアイデアで面白い」、「目的とそれに対するアプローチが良かった」などアイデアについてポジティブなコメントが多かった。各グループそれぞれ「楽しめるシステム」を目標として制作を行った。発表時にデモを交えて説明を行ったことで各システム面白さや楽しさをよりわかりやすく伝えることができたため、聴衆を楽しませるという目標は達成できたといえる。しかし、最終発表会時点では結論や評価について検討中であったため、それについての指摘も多く見受けられた。そのため、本プロジェクトの発表は完全と言い切れるものではなかった。

(※文責: 米森麻翔)

### 12.4.4 成果発表会後の自己評価

前述のとおり、発表技術について、プロジェクトについての評価点はいずれも平均7点を超えていた。10点満点において7割以上の評価を得られたということでおおむね高評価であったといえる。コメントも「わかりやすい」「面白い」などポジティブな声が多かった。しかし、各班、結論が検討中である点や声の大きさについて何件か指摘されていた。これらのことから、内容や説明はわかりやすいにもかかわらず、声が小さいばかりに十分な理解を与えることができなかったことがわかる。しかしながら、十分な発表には至らなかったものの発表会に向けての準備段階では全員が協力して発表準備を行っていた。そのためチームワークは非常に良かったといえる。以上を踏まえ

## X-Reality and Smartwatch Connecting Your Feelings and the World

て本プロジェクトの発表は自己評価として5段階評価中4点とした。

(※文責: 米森麻翔)

## 第 13 章 まとめ

本プロジェクトは VR とスマートウォッチを利用して「自分の気持ちと世界をつなぐ」ことを目標とするプロジェクトである。目標達成のために前期で習得した技術の応用方法を模索する必要がある。後期ではグループを再編し 3 班に分かれて目標達成につながる成果物の作成を目標として活動し、A 班では「おうち世界旅行」、B 班では「恐怖症診断」、C 班では「ユーザの睡眠を見直すシステム」をそれぞれ開発した。

(※文責: 米森麻翔)

### 13.1 各グループのまとめ

この節では、後期における各グループの活動と今後の課題を述べる。

(※文責: 米森麻翔)

#### 13.1.1 A 班のまとめ

本グループでは運動データを収集し、そのデータを活用して「楽しさ」を可視化するため、映像を眺めながらトレッドミルの上を歩いて疑似観光体験をしてもらい、スマートウォッチを用いて運動データを収集するために街並みの 3D モデルと、得られた運動データに対して重回帰分析を行いユーザの楽しさの度合いを推測するプログラムを開発した。映像技術は結城、スマートウォッチに関する技術は新田、解析に関するプログラムは米森がそれぞれ担当した。3D モデルは、最終発表段階では秋葉原と札幌市狸小路商店街の二種類を作成した。ソフトウェア名が「おうち世界旅行」であるにも関わらず、最終発表会段階では前述二種類といずれも日本国内の観光地のみで開発で終わってしまった。そのため今後は世界各地の観光地や有名スポットなど様々な 3D モデルを実装する必要がある。ユーザの楽しさを推測するにあたって、重回帰分析を行った。重回帰分析は Jupyter Notebook を用いて Python 3 で行った。前期では理解度向上のためにアルゴリズムをプログラムで実装したものを、後期ではライブラリを有効活用しプログラムを短縮することができた。重回帰分析に用いた学習データについて、歩行運動時の時間、ペース、被験者の最大心拍数を計測した。しかし学習データ量が十分ではなく、一部ダミーデータを用いていたためユーザの楽しさを推測するプログラムは完全とは言えない結果となった。そのため、今後はより多くの学習データを収集し、解析の精度を上げていく必要がある。

(※文責: 米森麻翔)

#### 13.1.2 B 班のまとめ

本グループでは恐怖症診断を実現するため、VR ゴーグルで恐怖症に対応する映像を視聴した際の心拍数を計測するアプリケーション、アプリケーションから得られた心拍数の時系列データをも

とに、統計学的手法で解析した結果から、ユーザが恐怖症映像に対してどの程度の恐怖を感じたかを推測するプログラムの二つを開発した。アプリケーションの開発は品川、松本が担当した。プログラムの開発は高杉が担当した。アプリケーションの開発では高所、深海、先端の計3つの恐怖症に対応する映像を作成した。また、恐怖症映像を視聴している際の心拍数をスマートウォッチで計測するため、心拍数受信・書き出しシステムの改良を行った。映像の作成について、開発環境の構築に時間がかかったため、映像の作成に時間を割くことができず、三つの恐怖症映像の作成に留まってしまった。また、恐怖症映像に恐怖を感じさせる工夫をあまり施すことができなかった。今後は広い範囲の恐怖症を診断を目指し、より多くの恐怖症映像を作成していく必要がある。また、恐怖症のクオリティを向上させるために、映像構成を検討していくべきである。スマートウォッチでの心拍数の計測は、計測からファイルに出力までの機能が実装できたため、計測部分については十分に開発を行えた。解析プログラムの開発では、恐怖症映像を視聴した際の心拍数の時系列データを計測し、学習データの作成を行った。また、Pythonで学習データに移動平均フィルタを適用するプログラム、主成分分析を行うプログラム、マハラノビス距離によるパターン認識を行うプログラムの開発を行った。実際に開発したアプリケーションを用いて、恐怖症映像視聴時の心拍数の時系列データを収集し、クラス分けを行ったものを学習データとして解析を行った。解析結果から、心拍数の時系列データからユーザが恐怖症映像に対してどの程度の恐怖を感じたかの推測が可能であると考えられた。しかし、恐怖症映像に対する恐怖度の推測に留まってしまい、当初の目標である、恐怖症診断の実現はできなかった。また、今回の解析には、データ数の不足、計測データの有意性が示せていないなどの課題がある。恐怖症診断実現のために、恐怖症映像視聴時と、それ以外の映像視聴時の心拍数の比較を行い、計測データの有意性を確保する。計測対象の再検討を行う。心拍数以外の身体データにも注目し、などを行っていく必要がある。それぞれのデータが恐怖の情動とどのような関係があるのかを精査していく。以上のような取り組みをしていく必要がある。

(※文責: 高杉拓海)

### 13.1.3 C班のまとめ

本グループでは、アプリケーションを開発し、ユーザーの睡眠の促進・目覚めの改善・睡眠の見直しを目的とする。その目的を達成するために、前期の三つのグループ活動を生かし、スマートウォッチや活動量計からリアルタイムで取得した心拍数を用いたアプリケーション開発をした。映像技術は長澤、スマートウォッチに関する技術は川端、解析に関するプログラムは丹下がそれぞれ担当した。アプリケーションの開発では、睡眠時のシーンで睡眠を促進させる背景とBGMを表示する予定であった。今回は時間不足により、促進させる効果が証明されていない背景を仮で表示させておいた。心拍数を参照する最低限の機能は完成できたが、睡眠促進の根拠のある背景や、レム睡眠時の心拍数の自動計測などの実装に至ることができなかった。スマートウォッチからUnityへ心拍を送信するようにし、心拍数をファイルで表示させるようにした。これを用いて、アプリ内でのアラーム機能の実装・解析を行った。また、Pythonを用いて移動平均フィルタ、主成分分析、パターン認識とそれぞれの結果をファイルに出力するプログラムを作成した。解析をした結果、心拍数で浅い、深い睡眠が区別できている可能性があるということがわかった。しかし、学習データの収集を始めるのが遅れてしまい、データが不足してしまったため、データ中にダミーデータが含まれており、結果の信頼性には注意する必要がある。より睡眠の促進、目覚めの改善、睡眠の見直し

しの精度を高くするためには、データ数の増加や、そのデータが有意なものであるかの検証を行うこと、スマートウォッチから得られる心拍数以外の身体データにも着目していく必要がある。

(※文責: 川端勇雅)

## 13.2 プロジェクトのまとめ

この節では、プロジェクトのまとめについて述べる。

(※文責: 丹下裕太)

### 13.2.1 プロジェクトのまとめ

本プロジェクトは、VR とスマートウォッチを用いて、「自分の気持ちと世界をつなぐ」情報リンクの作成を目的とした。目的達成のために、前期では映像班、SW 班、解析班の三つに分かれ、それぞれ技術習得を行った。そして、成果物として3班の技術を統合し爆弾ゲームと感情診断を作成した。後期ではグループを再編し、3班に分かれて目的達成のために成果物を作成した。A 班は運動データ、B 班は恐怖症、C 班は睡眠データに注目してそれぞれ活動を行った。本プロジェクトの目的について、各班がスマートウォッチから得られる身体データを用いて「自分の気持ちと世界をつなぐ」ための情報リンクを作成することは達成できたと考える。しかし、データ数不足や、データの信頼性、有意性の検証を行う必要があるため、完全に達成できてはいないと考える。

(※文責: 丹下裕太)

## 13.3 反省と今後の展望

この節では、各メンバが後期の活動を通して見つけた反省点と改善策について記述する。

(※文責: 米森麻翔)

### 13.3.1 各メンバの反省

メンバをさらに3つのグループに分けたことで、それぞれの負担が大きく異なることを認識するのに遅れが生じたことが反省点である。要件定義の段階で多くの時間を費やしてしまったために、スケジュールの余裕がなくなっていき、特に映像技術担当のメンバの負担が多くなってしまったことから、成果物の実装に多くの手間を要した。よって、分担時に生じる負担の偏りを考慮に入れていなかったことが反省であった。

(※文責: 新田蒼心)

成果発表会まではおおむね計画通り作業を進めることが出来たが、報告書などの期末提出物を作成する際に必要なものの確認を怠っており、計画通りに進まないことが多かったため、反省する点が多かった。

(※文責: 川端勇雅)

## X-Reality and Smartwatch Connecting Your Feelings and the World

やはり技術的に優秀な人の負担が大きくなってしまったなど感じた。円滑に進めるための方法として仕方無いが、もう少し負担を分けられるように出来たなど反省。改善策としては計画を細かくしっかりと立てることかなと思った。

(※文責: 品川誠剛)

スケジュール管理が上手くいかず、目標としていた映像数に届かなかった。映像制作では段取りが上手くいかず、映像の構成や流れをしっかりと決めていなかったため、十分なクオリティではなかった。

(※文責: 松本柊哉)

プロジェクト自体の進め方等を話し合う機会において、もっとよいやり方があったのかもしれないと思う。また、前述のとおりではあるが、進め方を確立してから全ての作業に着手することで、よりよい成果物を作れたのではないと思う。自分自身のモチベーションを維持するためにも、積極的に自分の意見を発信する必要があると思う。

(※文責: 結城熙乙)

教員、メンバ間でのコミュニケーションが不足しており円滑に作業を進めることができなかった。もっと積極的に話し合いや質問を行えば作業の滞りも少なくできたと感じる。また、グループ内でのコミュニケーションも不足しており、成果物のクオリティや必要な技術の相互理解に時間がかかってしまったために理想とする完成度に届かなかった。

(※文責: 米森麻翔)

教員とのコミュニケーション不足で、解析方法の課題点に気が付くことができず、プロジェクトの進行を遅延させてしまった。滞りなく作業を行うことができているならば、目標とした恐怖症診断を実現することができたと感じた。積極的に教員へ質問しに行くべきだった。また、教員が自身の質問の意図を理解しやすくなるように、自身が直面した問題点を事前にまとめておくべきだった。

(※文責: 高杉拓海)

教員、メンバ間でのコミュニケーションが不足し、問題点の発見の遅れや、作業の滞りが発生した。気になることがあればすぐに教員に話をしたり、メンバ間で積極的に情報を共有するべきであった。

(※文責: 丹下裕太)

自分の作業のことで頭がいっぱいになって、ほかの人の進捗の確認を怠ってしまっていた。ほかの人が何をやっているか、どんな技術を使用しているかなどを把握することができなかった。そして、発表会の準備にもあまり携われなかった。ある程度余裕を持って自分の作業の計画を立てたほうが、ほかの仕事を手伝えることができプロジェクトの進行がうまくいくことを学んだ。

(※文責: 長澤颯音)

## 参考文献

- [1] ユニティちゃんが教える！初心者向け Unity 講座  
<https://www.udemy.com/course/unity-chan-tutorial-01/learn/lecture/9598986#overview>,  
Mar.2018 (accessed Oct.2023).
- [2] 開発者の登録を行い、開発者モードをオンにする  
<https://website-note.net/vr/oculus-quest2-how-to-regist-dev/>, Jan.2021 (accessed  
Jun.2023).
- [3] 動画ファイルから連番画像を作る  
<https://qiita.com/fuk101/items/cf4db17419ff913cb9b2#ffmpeg>, Aug.2022 (accessed  
Jul.2023).
- [4] じん, ”心拍数を配信画面に載せる方法!!,” <https://tokaisodachi.com/archives/2335>, Jun.2020  
(accessed Jul.2023).
- [5] 期末報告書 (修正版), google drive, 2023 年 7 月.
- [6] 主成分分析資料, 公立ほこだて未来大学佐藤仁樹研究室資料, 2023 年 5 月.
- [7] 重回帰分析資料, 公立ほこだて未来大学佐藤仁樹研究室資料, 2023 年 5 月.
- [8] パターン認識資料, 公立ほこだて未来大学佐藤仁樹研究室資料, 2023 年 5 月.
- [9] Unity Technologies, ” AllSky Free - 10 Sky / Skybox Set”  
<https://assetstore.unity.com/packages/2d/textures-materials/sky/allsky-free-10-sky-skybox-set-146014>, Aug 18. 2021(accessed Dec.2023).
- [10] Unity Technologies, ” Dropdown” <https://docs.unity3d.com/ja/2018.4/Manual/script-Dropdown.html>, Apr.2021 (accessed Dec.2023).
- [11] 運動強度とは  
<https://www.tyojyu.or.jp/net/kenkou-tyoju/shintai-training/undou-kyoudo.html>, Jun 1.  
2023 (accessed Dec.2023).
- [12] The Blender Foundation, <https://www.blender.org/>, (accessed Jan.2024).
- [13] ffish.asia / floraZia.com, “アカシユモクザメ Scalloped Hammerhead Shark, S. lewini,” ,Sketchfab, <https://sketchfab.com/3d-models/scalloped-hammerhead-shark-s-lewini-b68fdc989ba74bec9495ac907995739e>, Oct. 2022 (accessed Jan.2024).
- [14] 八十政夫, “心拍数の分散値を用いた REM 睡眠の判別,” 電気関係学会, 鹿児島大学, 2004 年 9 月.
- [15] Microsoft, <https://visualstudio.microsoft.com/ja/vs/whatsnew/>, (accessed Jan.2024).
- [16] Python Software Foundation, <https://www.python.org/>, (accessed Jan.2024).
- [17] Numpy で正確な移動平均を取る方法, <https://zenn.dev/bluepost/articles/1b7b580ab54e95>,  
Nov 7. 2021 (accessed Jan.2024).
- [18] Creative Commons, <https://creativecommons.org/licenses/by/4.0/>, (accessed Jan.2024).

## 付録 A 使用した技術について

この章では、これまで我々が学習したもののうち、期末成果物で用いた技術について、詳細に解説していく。

(※文責: 結城熙乙)

### A.1 Unity

#### A.1.1 概要

Unity とは、米 Unity Technologies が開発したゲームエンジンである。

- C# で書かれたスクリプトを、ゲーム内のオブジェクトにアタッチすることができる。
- 主に Unity Editor という開発環境が用いられる。

(※文責: 結城熙乙)

#### A.1.2 成果物への関与

当プロジェクトの成果物における、全ての映像に Unity が用いられている。

(※文責: 結城熙乙)

### A 班

- 心拍数とカメラ速度の連動は、後述の Node.js サーバと Unity 間で WebSocket 通信を行うことで実現している。
  - まず、WebSocketSharp という Unity で WebSocket 通信を行うためのライブラリを用いることで、Node.js サーバ から心拍数を文字列型として Unity が受け取る。この過程を経た後に、文字列型から浮動小数点数値型に型変換を行う。そして、数値型になった心拍数に応じた値を、プレイヤーオブジェクトにアタッチされた Cinemachine Dolly Cart の速度に乗算することで実現している。なお、この処理を C# スクリプト内の Update メソッドに記述することで、リアルタイムの心拍数取得を実現している。
- OBJ 形式である先述の PLATEAU を直接取り込むと、座標のずれが起きてしまったので、一度 Blender にて修正を行った。その後、Blender から FBX 形式にてエクスポートした後に Unity へインポートした。
  - Blender: 3D モデリングソフトウェア

(※文責: 結城熙乙)

## B 班

- Unity に 360 度動画を取り込むために、SkyboxPanoramicBetaShaderGUI.cs を取り込んだ、それにより 360 度動画を正距円筒図法で取り込むことが可能になり、Skybox/Panoramic シェーダーに 360 度動画のテクスチャを張り付けることにより、Unity での 360 度動画の再生を可能にした。その際に参考にした資料が英語で書かれてたので、日本語訳に書き換えた資料を作成した。

(※文責: 長澤颯音)

## C 班

- 心拍数を条件に追加したアラームの設定を行うために、DateTime 構造体、PlayerPrefs, Dropdown などの Unity の機能を用いてアプリを作成した。詳しい方法は第 11 章 4 節を参照。

(※文責: 長澤颯音)

## A.2 ANT+

### A.2.1 概要

ANT とは、米 GARMIN 社傘下のカナダ企業である、Dynastream Innovations inc. が開発した無線センサネットワークの独自プロトコルである。この ANT をベースとした通信互換プロトコルが ANT+ である。ANT+ は、ANT+ Alliance が管理運営している。主にサイクルコンピュータ、SW、心拍センサと他のコンピュータやスマートフォンの通信に用いられている。

(※文責: 結城熙乙)

### A.2.2 成果物への関与

成果物においては、ANT+ 規格対応の SW(GARMIN VivoSmart) と、PC で ANT+ 通信を行うための USB ドングルを用意し、SW から PC 間のリアルタイム心拍数データ送信を実現している。Windows PC で ANT+ の USB ドングルを用いるためには、WinUSB ドライバをインストールし、USB ドングルで使用する設定が必要だった。これを行うために、Zadig というソフトウェアを用いた。

(※文責: 結城熙乙)

## A.3 Node.js

### A.3.1 概要

Node.js とは、JavaScript 実行環境の一つである。Node.js は、アメリカのプログラマである Ryan Dahl によって開発された。現在は彼を雇用している米 Joyent 社によって支援されている、

オープンソースソフトウェアである.

(※文責: 結城熙乙)

### A.3.2 成果物への関与

本プロジェクトの全ての成果物に Node.js が用いられている.

(※文責: 結城熙乙)

## A.4 Python

### A.4.1 概要

Python とは, 1991 年に Guido van Rossum によって開発されたプログラミング言語である.

(※文責: 高杉拓海)

### A.4.2 成果物への関与

当プロジェクトの前期成果物と各班の後期成果物における, 全ての解析を行うプログラムに Python が用いられてる. 以下に前期成果物と各班の成果物で用いた Python のライブラリ・モジュールについて示す

(※文責: 高杉拓海)

#### 前期成果物・B 班・C 班

**numpy** 様々な数学関数, 乱数生成器, 線形代数ルーチン, フーリエ変換を提供するライブラリ.

**pandas** csv ファイルの読み込み, 書き込みを可能にするライブラリ.

**collections** Python の標準ライブラリ. dict, list, set, tuple 型に代わる特殊なコンテナデータ型を実装している.

(※文責: 高杉拓海)

#### A 班

**pandas** データ解析のためのライブラリ

**sklearn.linear\_model** データ分析や機械学習のためのライブラリである scikitlearn のうち線形回帰による予測を行うモジュール

**sklearn.preprocessing** データ分析や機械学習のためのライブラリである scikit-learn のうちデータの前処理、特に正規化や標準化などを行うモジュール

**sklearn.metrics** 作成したモデルの評価を行うモジュール

(※文責: 米森麻翔)