

5 Leap Motion を用いた迷路ゲームの作成

工藤裕貴

指導教員 ソソラ

1. はじめに

手の動きを検知してコンピュータを操作することが可能なデバイスである「Leap Motion」に興味を持った。

そこで Leap Motion で手の傾きを取得し、その傾きを使って遊べるアプリとして迷路ゲームを作ろうと思い、このテーマを設定した。

2. 研究概要

2.1 目的

本研究の目的は Leap Motion で操作する迷路ゲームを作成することである。

以下に完成イメージを示す。



図 1 完成イメージ

2.2 開発環境

OS	Windows7/8.1
ゲームエンジン	Unity 5.4.0 f3
使用機器	Leap Motion
使用言語	C#

2.3 Leap Motion とは

2012 年に発売された手のジェスチャーによってコンピュータの操作ができるデバイス。

USB で接続することで使用できる。

検知できる範囲は半径 50 センチ、中心角 110° で 1/100mm の精度で認識できる。

フレーム単位で最高 200fps の間隔で動作し、手の座標情報は図 2 のような手の関節の骨の座標を読み取る。

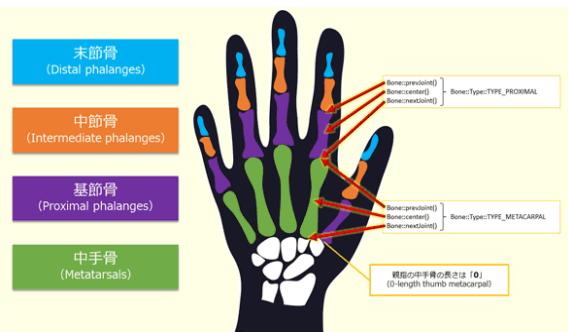


図 2 手の関節の骨

2.4 Unity とは

統合開発環境を内蔵し、複数のプラットフォームに対応するゲームエンジン。

主にモバイルやブラウザゲーム製作に使用される。

2.5 3D ホログラムについて

本研究では当初、ゲームを 3D ホログラムでの投影を考えていたが、現時点では PC のディスプレイに表示するアプリを制作している。

今後ゲームが完成し、時間に余裕があればゲームを改良し投影を考えたい。

3. 迷路ゲームについて

ステップ 1. 難易度別のコースからコースを選択する。一番簡単なものは 1 本道にする。

ステップ 2. Leap Motion によりディスプレイ上のコースを図 1 のように手の傾きで前後左右に

傾けて操作し,ボールを転がしながらゴールまで移動させる.

傾きのアルゴリズム

- ① 両手の中指と手のひらの座標を取得する.
- ② 中指の 2 点の座標平均と手のひら 2 点の座標平均を求める.
- ③ それぞれ求めた座標から図 3 の公式を使い,前後の傾きを求める.
- ④ 同様に手のひらの 2 点の座標から図 3 の公式を使い,左右の傾きを求める.

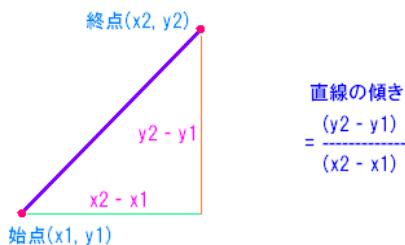


図 3 傾きの公式

- ⑤ 前後と左右それぞれ傾きの値が正か負かにより傾ける向きを判断する.
- ⑥ 4 で判断した向きに前後左右に傾ける.
この時コースが裏返るのを防ぐために傾きに角度制限を付けて,範囲からはみ出すときは傾けない.

4. 実験結果

実験 1

Unity のチュートリアルの玉転がしゲームを拡張し,キー操作で直接ボールを動かす迷路ゲームを作成した.

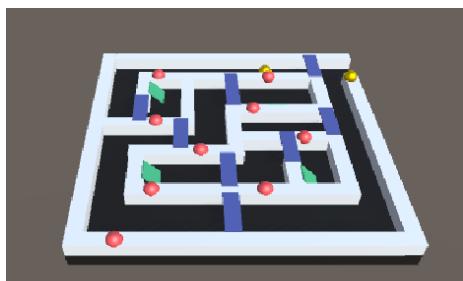


図 4 作成した迷路ゲーム

実験 2

研究概要でも説明したように Leap Motion は手

の座標情報を関節の骨単位で認識している.それを学習するために指先の座標を取得し,球に反映させるサンプルプログラムを拡張し,手のひらとすべての骨の座標を取得するようにした.

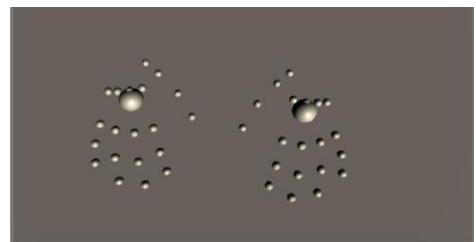


図 5 手の座標を取得するプログラム

5. 終わりに

進捗状況

Unity を学習し実験 1 により迷路ゲームの原型となるものが作成できた.

実験 2 のプログラムにより,Leap Motion により手を認識し,手の関節の骨ごとの座標を読み取ることができた.

迷路コースを傾ける際の角度の制限を付けるための処理を完成させた.

現在は手の傾きの処理を作成している.

今後の課題

- 手の傾きの処理の完成と調整
- 難易度別のコースの作成:難易度が高いものには行き止まりの道や,落とし穴等のギミックを追加する予定.
- ゲームを 3D ホログラムで表示することの検討

今後の作業計画

12月	ゲームの完成・改良
1月	
2月	発表資料作成

6. 参考文献

Leap Motion

<https://www.leapmotion.com/?lang=jp>

Unity

<http://japan.unity3d.com/>