



VR型ドライブシミュレータの 設計

平石研究室 S18059 柏瀬 悠輔

目次

1. はじめに
2. ゲーム制作方法
3. 卒業研究Aの内容
4. 使用機材
5. OculusQUEST2の説明
6. LogicoolG29の説明
7. 開発環境
8. ゲーム内容
9. Level別の車の最高速度
10. 自車を操作するプログラム
11. 自車プログラムの説明
12. 障害物を配置するプログラム
13. 設定の手順
14. PCとOculusQUEST2の接続方法
15. Unityの設定
16. 実行方法
17. 実行動画
18. おわりに
19. 参考文献

はじめに

本研究では、VR型のドライブシミュレータの作成を目的とする。
VR技術を利用することで、実際の運転ではなく、シミュレーション
環境による現実に近い形での実験を可能にする。
作成したドライビングゲームをゲーム用のドライビングコントローラに
対応させ、VR 装置への拡張を行なった。

ゲーム制作方法

ドライビングゲームを制作するためにUnityを使用した。
ゲームに使用する車や道路(図2)はアセットストア(図1)からインストールした。

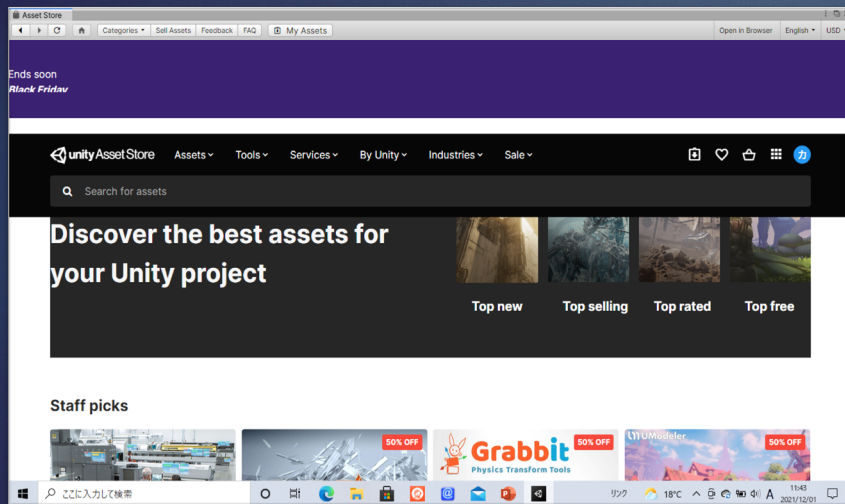


図1:unityのアセットストア

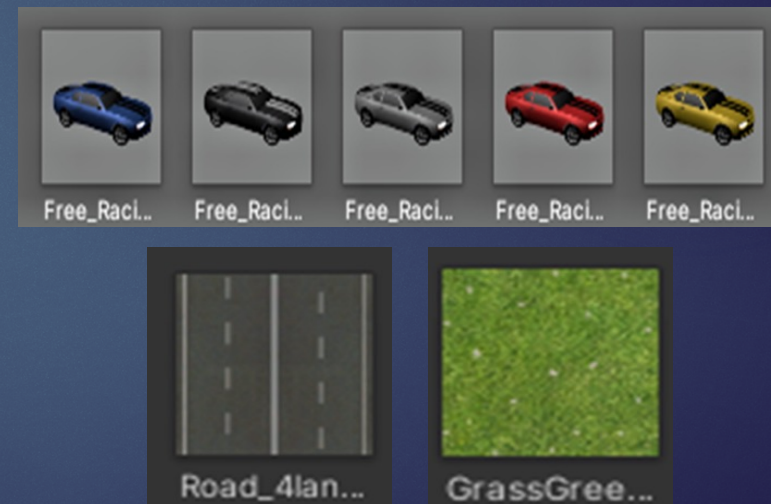


図2:ゲーム素材

卒業研究Aの内容

卒業研究Aでは、ゲームエンジンであるUnityを利用し、簡単なドライビングゲームを作成した。

他の車をランダムに配置し、PCのキーボードを使って自車を操作し、他の車との衝突を避けながら、移動することを可能にした。

使用機材

- Oculus QUEST2(図3)
- Logitech G29のドライビングコントローラ(図4)



図3:Oculus QUEST2



図4:ドライビングコントローラ

OculusQUEST2の説明

本研究では、ドライビングゲームをVRに拡張するため
OculusQUEST2を使用した。

OculusQUEST2は、Unityに対応しており、Unityで
Oculus Integrationというライブラリをダウンロードすることで、
作成したアプリケーションをVR環境に対応させることが可能である。

LogicoolG29の説明

本研究では, Logicool社製のDRIVING FORCEレーシング
ホイールG29を採用した.

この機種はOculusQUEST2と同様でUnityに対応しており,
Logitech SDKを利用して独自のプログラムに, ドライビング
コントローラを対応させることが可能である.

開発環境

- Windows10 使用しているPC
- Oculus Integration Unityのプロジェクトに取り込む
- Logitech SDK Unityのプロジェクトに取り込む
- Unity 2020.3.25f1 ゲーム開発エンジン

ゲーム内容

本研究のドライビングゲームは、障害物の車に当たらないように直線の道路を走る簡易的なゲームになっている。

障害物に衝突せずに、制限時間が0になるとGAMECLEARという文字が画面に表示されてゲームが終了される(図5)。

障害物に衝突した場合には、その場でGAMEOVERとなる(図6)。



図5:GAMECLEAR画面



図6:GAMEOVER画面

Level別の車の最高速度

本研究のドライビングゲームの最初の制限時間(Time)は45秒に設定しており, Levelが上がるにつれて10秒ずつ制限時間が長くなるように設定した.

また, Levelが上がるごとに車の速度が速くなる(表1).

表1:最高速度の変化表

Level	最高速度(m/s)
Level 1	50
Level 2	75
Level 3	100
Level 4	125
Level 5	150

自車を操作するプログラム

1行目のrec.lXは

ハンドル操作の値 (steer).

2行目のrec.lYは

アクセルペダルの値 (accel).

3行目の rec.lRzは

ブレーキペダルの値 (brake).

rec.rglSlider[0]ハンドブレーキの
値 (hand)を取得するための変数で
ある.

それぞれ、符号付き整数2バイト
の範囲(-32768から32767)で

データを取得することができる.

```
1 steer = rec.lX / 32768f;
```

```
2 accel = -1 * rec.lY / 65536f + 0.5f;
```

```
3 brake = -1 * rec.lRz / 65536f + 0.5f;
```

```
4 hand = -1 * rec.rglSlider[0] / 65536f  
+0.5f;
```

自車プログラムの説明

ハンドルの値は、左に限界まで切った状態で-1, 真ん中で0, 右に限界まで切った状態で1となるようにしている.

2行目から4行目のアクセルの値, ブレーキの値, ハンドブレーキの値では, それぞれの変数の値が, 踏んでいない状態でプラス, 踏んだ状態でマイナスとなるため,

踏んでない状態で0, 強く踏んだ状態で-1となるように, それぞれの値をセットしている.

```
1 steer = rec.IX / 32768f;
2 accel = -1 * rec.IY / 65536f + 0.5f;
3 brake = -1 * rec.IRz / 65536f + 0.5f;
4 hand  = -1 * rec.rglSlider[0] / 65536f
          +0.5f;
```

障害物を配置するプログラム

右側のプログラムは障害物をランダムに配置するプログラムの一部である。

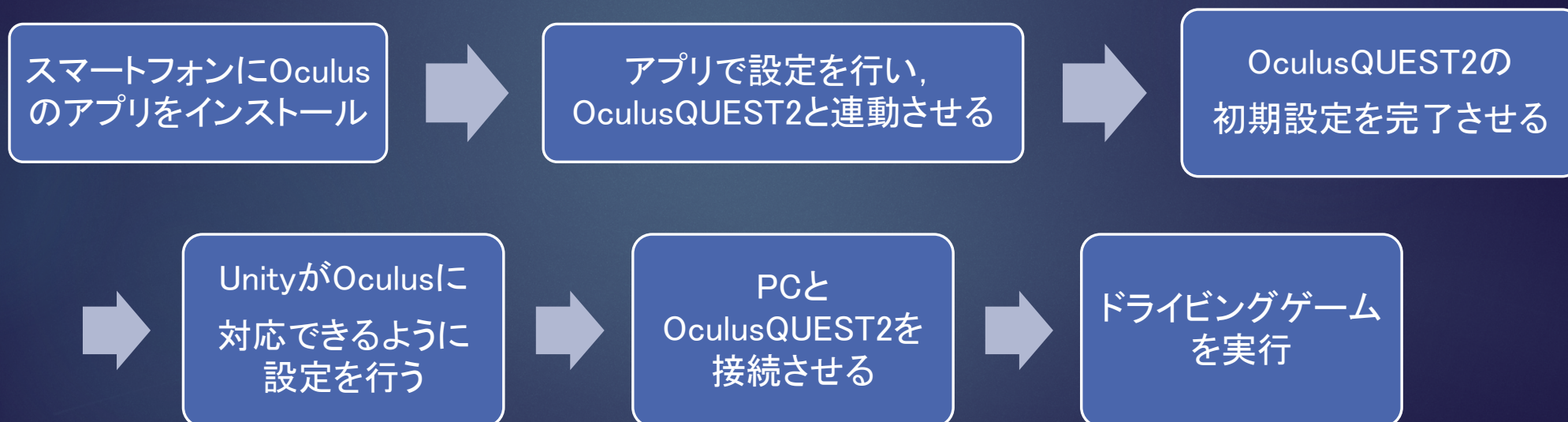
1行目から4行目で、他の車のオブジェクトを4つ生成している。

6行目から8行目の処理によって、自分の車が座標値で80だけ移動するごとに、他の車をランダムに生成している。

```
1  public GameObject enemy1;  
2  public GameObject enemy2;  
3  public GameObject enemy3;  
4  public GameObject enemy4;  
5  void Update() {  
6      if (transform.position.z > 80) {  
7          CreateEnemy();  
8      }  
9  }
```

設定の手順

OculusQUEST2をUnityで使用するために以下の手順で設定を行った。



PCとOculus QUEST2の接続方法

PCとOculus QUEST2を接続するためにUSB Type-Cケーブルを使用した。

PCとOculusQUEST2をケーブルで接続すると図7のようにPCのUnity画面をOculusQUEST2で見ることが可能になる。

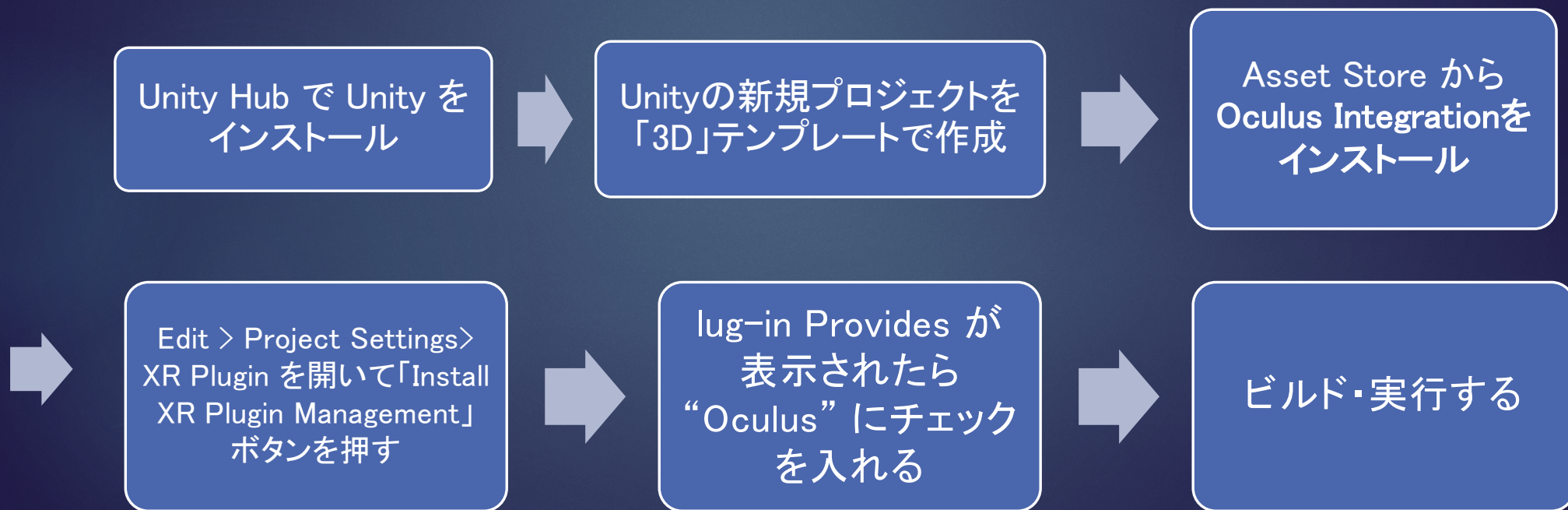


図7:接続後の画面

Unityの設定

OculusQUEST2でUnityの画面を見るためにUnityで設定を行う。

設定の手順は以下の通りである。



実行方法

PC側のUnityの実行ボタンをクリックすると、OculusQUEST2でドライビングゲームの実行画面を見ることが出来る。

次のスライドの実行動画は指定された50MBのサイズ制限を超えないように動画を圧縮している為、画質が少し粗くなっています。

実行動画



おわりに

本研究では、Unityで作成したドライビングゲームをドライビングコントローラに対応させ、VR装置への拡張を行なった。

設定を完了するためにダウンロードするものが多く、時間がかかるため環境が整った場所で作業をする必要がある。

参考文献

コース作成方法

<https://hiyotama.hatenablog.com/entry/2015/05/05/080000>

追跡カメラコード

<https://codegenius.org/open/courses/24/sections/118>

衝突したらゲームオーバーシーンに切り替えるコード

https://futabazemi.net/notes/unity_scene/

G29ハンドルをUnityで使う方法

<https://fujiya228.com/unity-logitech-g29-setup>

OculusQUEST2をUnityで使用方法

https://manurito.com/3d_vrdev01/

<https://framesynthesis.jp/tech/unity/oculusquest/>