

Unity での校内紹介アプリの制作

井戸本太陽 大中 優佑
佐藤 悠惺 高田 明夢
金谷 晴琉 田邊 陸玖
岡本 真宙



図1 THETA SC2

1. まえがき

昨今、コロナの流行により、学校では、多くの行事が延期されたり、中止になったりしている。学校のいろいろな活動を人々に知ってもらう方法はないかと考えているとき、WebGLの技術を取り入れることによっていたって簡単に情報が発信できることを知ったのである。

2. 原理

UnityのWebGL機能を使ったアプリケーションの作成では、「進むボタン」や「戻るボタン」を全天球上に配置し、クリックすることで移動したり、マウスドラッグで画面を動かしたりして校内をあたかも散策できるようにしたものである。

3. 研究内容

(1) 使用機材の選定

全天球の背景部分にはりつける写真は360度前面を一度に撮影できるものが最適のためDJやINATA360など多くのカメラの中から今回は、リコーTHETA SC2(以後SC2とする。)を使用することにしたのである。これを使って、校内のいろいろな場所を撮影することにしたのである。図1は、SC2である。非常にコンパクトな本体とWifi、Bluetoothで簡単にスマホと連携ができるのである。

(2) 撮影した写真をUnityへ取り込む

SC2で撮影した写真素材をパソコンに転送し、図2下の特定のフォルダ(写真など)にインポートする。



図2 インポート

(3) sphere(球体)の設定

図3の右下のように、画像を全天球に張り付けられるように球体を作成する。

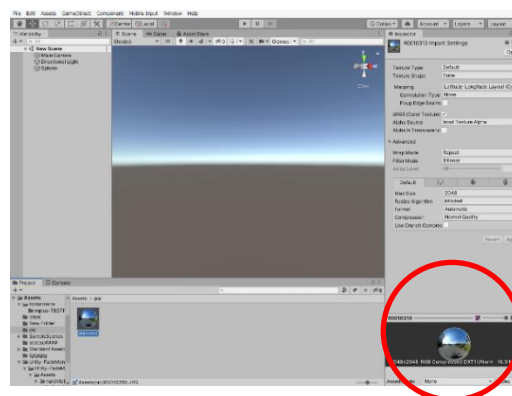


図3 全天球画像

(4) 球状のマテリアルを作成

図4左下のように、撮影した背景の写真を張り付けるためのマテリアルを作成する。

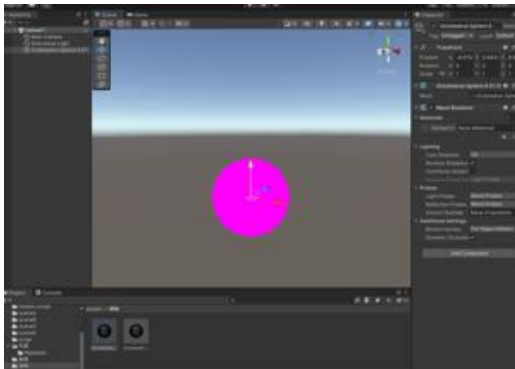


図4 マテリアル

(5) 画像を張り付ける作業

図5のように作成したマテリアルの内側の面に1枚1枚写真を張り付けていく。

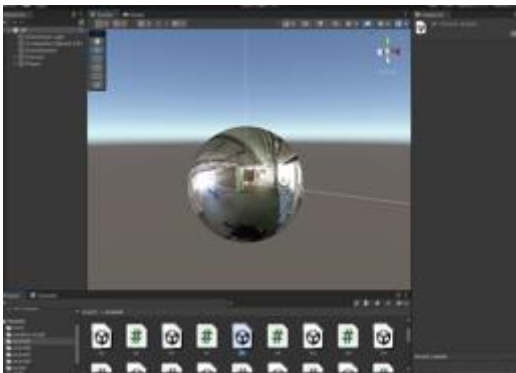


図5 画像貼り付け

(6) 視点移動のスキプットの作成

図6のカメラひとつひとつにマウスの左ボタンをドラッグする操作によりカメラの視点を変えることが可能であり、あたかも回転しているかのように見えるように工夫しているのである。

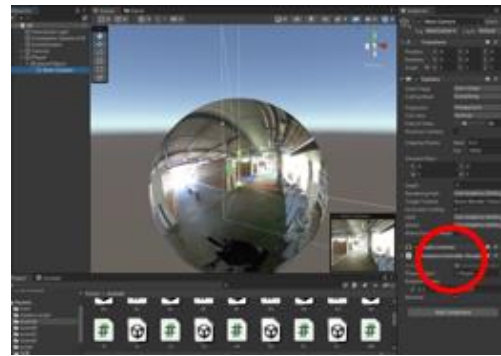


図6 カメラ

```
void Update()
{
    if (Input.GetMouseButton(0))
    {
        lastMousePosition = Input.mousePosition;
    }
    else if (Input.GetMouseButton(0))
    {
        if (!reverse)
        {
            var x = (lastMousePosition.x - Input.mousePosition.x);
            var y = (Input.mousePosition.y - lastMousePosition.y);
            if (Mathf.Abs(x) < Mathf.Abs(y))
            {
                x = 0;
            }
            else
            {
                y = 0;
            }
            var newAngle = Vector3.zero;
            newAngle.x = x * rotationSpeed.x;
            newAngle.y = y * rotationSpeed.y;
            mainCamera.transform.RotateAround(playerObject.transform.position, Vector3.up, newAngle.x);
            mainCamera.transform.RotateAround(playerObject.transform.position, transform.right, newAngle.y);
            lastMousePosition = Input.mousePosition;
        }
        else
        {
            var x = (Input.mousePosition.x - lastMousePosition.x);
            var y = (lastMousePosition.y - Input.mousePosition.y);
            if (Mathf.Abs(x) < Mathf.Abs(y))
            {
                x = 0;
            }
            else
            {
                y = 0;
            }
            var newAngle = Vector3.zero;
            newAngle.x = x * rotationSpeed.x;
            newAngle.y = y * rotationSpeed.y;
        }
    }
}
```

図7 カメラを動かすスクリプト

(7) 移動ボタンの作成

図8のそれぞれの画面（シーン）に配置された「進むボタン」や「戻るボタン」などを押すことによって希望するシーンへ移動することが可能となり、学校内の様々な場所に瞬時に行くことができるのである。

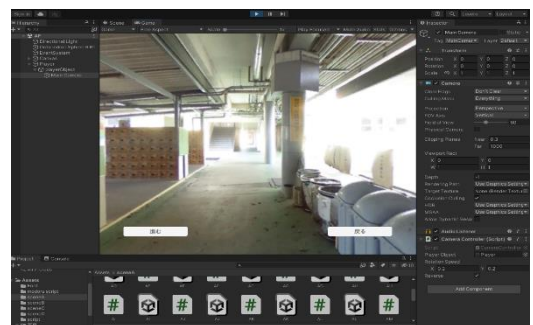


図8 移動ボタンの配置

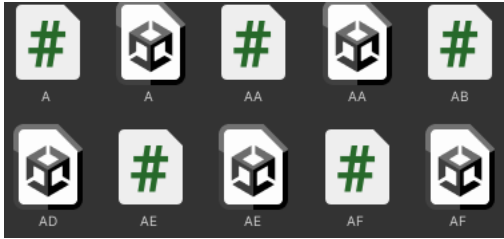


図8 移動シーンの作成

(8) シーンの設定

図9のように(3)~(7)を繰り返し撮影した写真の枚数を各移動するシーンを作成する。

```

using UnityEngine;
using UnityEngine.SceneManagement;

public class AA : MonoBehaviour
{
    public void OnClickStartButton()
    {
        SceneManager.LoadScene("AA");
    }
}

```

図9 シーン移動スクリプト

(9) 動画スクリプトの設定

図11のようにシーンないに作成したPanelに動画再生のスクリプトを設定(タッチ)する。

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.Video;

public class VideoScript : MonoBehaviour
{
    VideoPlayer video;
    // Start is called before the first frame update
    void Start()
    {
        video = GetComponent<VideoPlayer>();
    }
    // Update is called once per frame
    void Update()
    {
        if (Input.GetKeyDown(KeyCode.E))
        {
            if (video.isPlaying)
            {
                video.Pause(); // 動画を停止
            }
            else
            {
                video.Play(); // 動画を再生
            }
        }
    }
}

```

図11 動画のスクリプト

4. まとめ

この研究を通して改めて水工の伝統ある建物や雰囲気を感じることができた。

各々が得意な分野を生かして作業することができた。使ったアプリや機材の知識を知り、身に

着けることができた

昨年度からの引き続きの研究であったため多くの課題や実現が困難な問題が多々出てきたが、先輩方の研究のおかげをもちまして、自分たちの目標を立てることができ、よりよい成果が出たことに感謝いたします。

5. 参考文献

- **【Unity】** カメラ移動を制御するスクリプト
<https://qiita.com/Nekomasu/items/f195db36a2516e0dd460>
- Virtual Tour Using Sphere with unity
<https://www.youtube.com/watch?v=1Fupk75GhHg>
- Unity : ボタンクリックでシーン間を遷移(移動)する
<https://dianxnao.com/>
 ボタンクリックでシーン間を遷移(移動)する/
- Unityでのシーン切り替え演出について学ぶ
<https://bibinbaleo.hatenablog.com/entry/2017/10/08/161733>
- Wikipedia etc.