

# ペッパーズゴースト班 活動報告書

立命館コンピュータクラブ  
2017年度後期プロジェクト活動

2018年2月5日

程 瑞希<sup>1</sup> 青木 雅典<sup>2</sup> 中西 達也<sup>3</sup> 中西 将隆<sup>4</sup>  
藤田 朱門<sup>5</sup> 梁瀬 貞裕<sup>5</sup> 安田 直樹<sup>6</sup> 海木 一佳<sup>7</sup>

---

<sup>1</sup>理工学部機械工学科一回生

<sup>2</sup>理工学部電子情報工学科一回生

<sup>3</sup>理工学部建築都市デザイン学科一回生

<sup>4</sup>情報理工学部電子情報工学科二回生

<sup>5</sup>情報理工学部情報システム学科二回生

<sup>6</sup>情報理工学部情報メディア情報学科三回生

<sup>7</sup>情報理工学部情報システム学科三回生

# 目次

1	はじめに	3
2	活動概要	3
3	活動内容	4
3.1	三層スクリーンのペッパーズゴーストの仕組みについて	4
3.1.1	仕組み	4
3.1.2	ハードウェア	5
3.1.3	出力用の映像	5
3.1.4	実際に見たときの様子	6
3.2	三層スクリーンを生かしたメディアアートについて	7
3.2.1	概要	7
3.2.2	工夫	7
3.2.3	結果	8
3.2.4	お借りしたもの	8
3.3	Unity 上での設定方法	9
3.3.1	三層分の表示	9
3.4	三層スクリーンのスクリーン間移動について	10
3.4.1	スクリーン間移動の違和感	10
3.4.2	移動方法考察	10
3.5	三層スクリーンを利用したポケモン風 UI	12
3.5.1	内容	12
3.5.2	お借りしたもの	12
4	活動で得られたもの	13
5	展望	13

## 1 はじめに

文責：程 瑞希

ペッパーズゴーストとは、投影技術の一種である。その歴史は古く、1862年の舞台演出に使用されたのが始まりと言われている。現在ではディズニーランドのアトラクションやメディアアートとして社会に活用されている。MRやVRと違い、裸眼で直接見られることが利点である。本プロジェクトでは、そのペッパーズゴーストのスクリーンを一度に三枚重ねて出力したものを使うことで、新しい映像作品やゲームの形の開発を目指した。

## 2 活動概要

文責：程 瑞希

三層のペッパーズゴーストを投影できるハードウェアを一つ制作し、各班員が各自の得意分野で投影する映像の制作を行う形をとった。以下の活動内容に各班員の活動報告を記す。また、2017年度のRCCの学園祭では、個人製作物として展示した。

### 3 活動内容

#### 3.1 三層スクリーンのペッパーズゴーストの仕組みについて

文責：程 瑞希

##### 3.1.1 仕組み

45° に立てかけたスクリーンを三枚用意し、上から投影される映像を 90° に屈折させることで、三つの映像を同時に重ねて見ることができる。

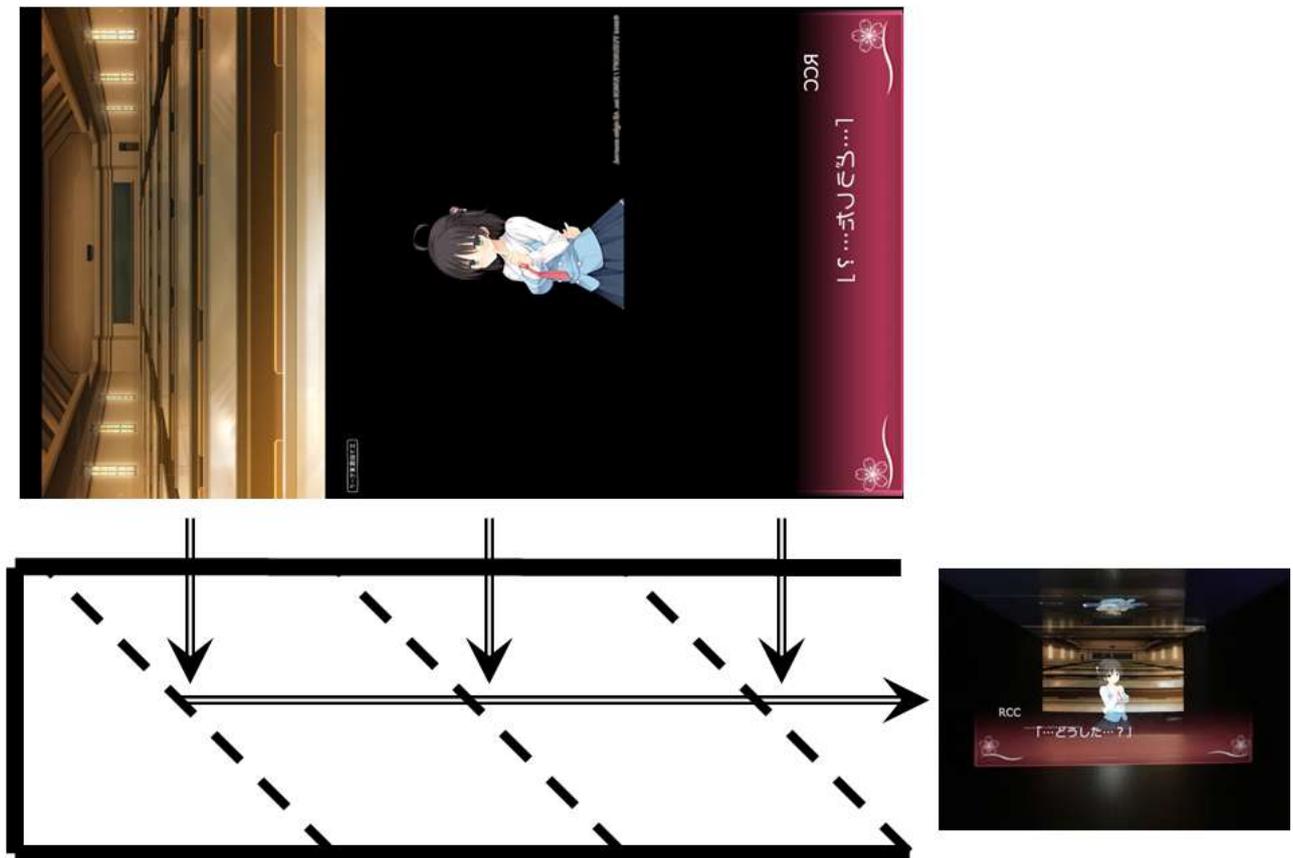


図 1: 側面から見た仕組みの図 (破線はスクリーン)( 2017 YUZUSOFT/JUNOS inc. all rights reserved.)

### 3.1.2 ハードウェア

投影された像がよく見えるように箱部分には黒色のプラスチックダンボールを使用した。光の位相のズレによって像が二重にならないようにスクリーンには薄い0.5mmの厚さの亚克力板を使用した。スクリーンが45°の角度でちゃんと固定されるように箱の内側に切れ込みを入れた黒いスポンジを接着した。この箱の上にノートパソコンの画面を設置して横から覗いて映像を見られるようにした。

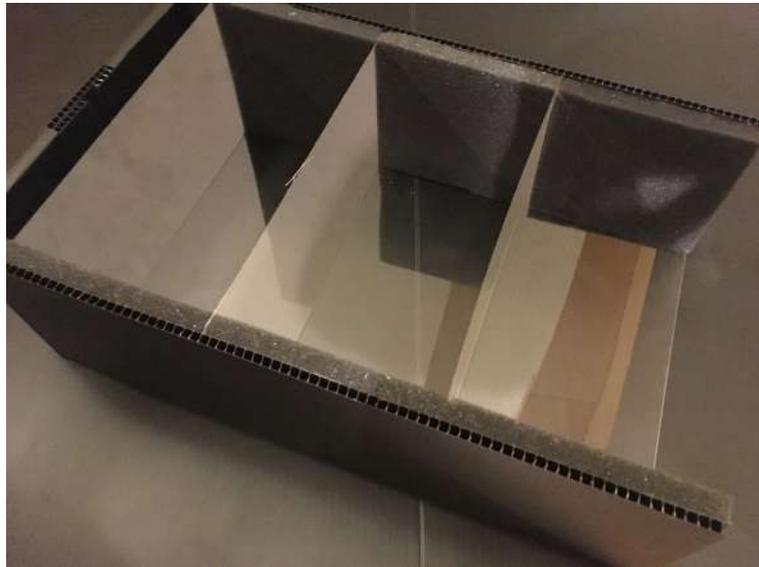


図 2: 制作したハードウェア

### 3.1.3 出力用の映像

三つの映像を同時に出力するために、下記画像のように9:16の比率の画面に16分の9倍の大きさにした9:16の映像を横向きに3つ表示させる形をとった。光の屈折の性質上、出力用の映像は左右反転させる必要がある。黒い部分は反射しないため、透過させたい部分は黒くする必要がある。



図 3: 出力用の映像の例 (2017 YUZUSOFT/JUNOS inc. all rights reserved.)

### 3.1.4 実際に見たときの様子

三つの映像が重なることで奥行きを感じられる画面が生まれる。また、視線をずらすことで1枚の画面では隠れてしまう後ろの背景などが見えるようになる。



図 4: 実際に見たときの例 ( 2017 YUZUSOFT/JUNOS inc. all rights reserved.)

## 3.2 三層スクリーンを生かしたメディアアートについて

文責：程 瑞希

### 3.2.1 概要

本プロジェクトの特色は三つの映像を三層のスクリーンに同時に出力できることである。よって MikuMikuDance を使ってその特色を生かした映像作品を作ることにした。

### 3.2.2 工夫

三層であることを生かすために、立体感を出す方法について考察した。奥のスクリーンに原曲の PV を、中央のスクリーンに踊るキャラクターを、前のスクリーンに歌詞などのエフェクトを投影した。立体感を出すために背景の PV には影を出力した。

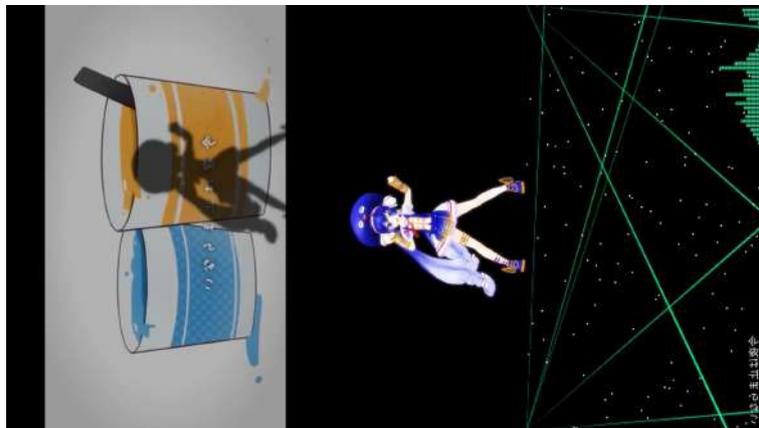


図 5: 出力用の映像

### 3.2.3 結果

キャラクターが立体的に見える新しい映像体験を生み出すことができた。

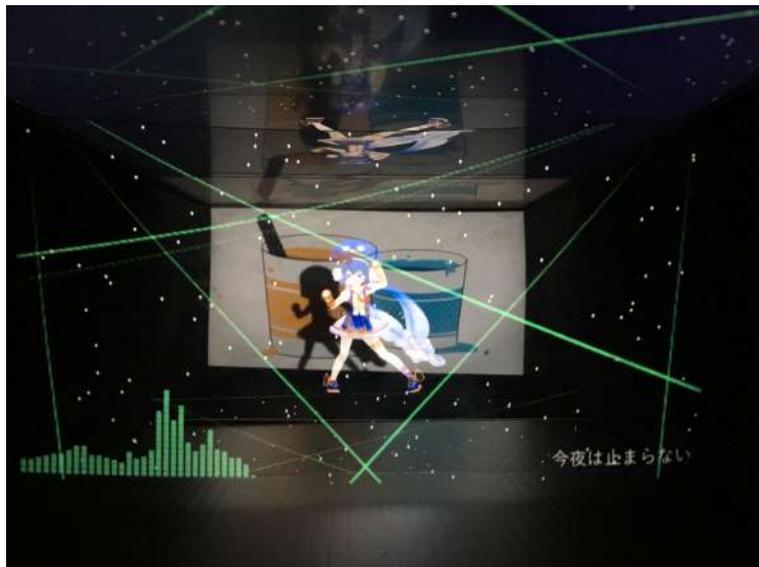


図 6: 実際の様子

### 3.2.4 お借りしたもの

- Model:<http://3d.nicovideo.jp/works/td27080>
- Motion:<http://www.nicovideo.jp/watch/sm18669204>
- PV:<http://www.nicovideo.jp/watch/sm17712448>

### 3.3 Unity 上での設定方法

文責：梁瀬 貞裕

#### 3.3.1 三層分の表示

MainCamera の ViewportRect の値を変更することによって三層スクリーン用の表示ができる。

	Y	H
奥	0.6328126	0.3164063
中	0.3164063	0.3164063
手前	0	0.3164063

また、このままでは左右が反転して見えてしまうので以下のスクリプトをそれぞれのカメラにつける。

```
1
2 using UnityEngine;
3 using System.ComponentModel;
4
5 public class ReverseRightAndLeft : MonoBehaviour
6 {
7     private Matrix4x4 mat;
8     private Camera cam;
9
10    void Start()
11    {
12        cam = GetComponent<Camera>();
13        mat = cam.projectionMatrix * Matrix4x4.Scale(new Vector3(-1, 1, 1));
14    }
15
16    void OnPreCull()
17    {
18        cam.ResetWorldToCameraMatrix();
19        cam.ResetProjectionMatrix();
20        cam.projectionMatrix = mat;
21    }
22
23    void OnPreRender()
24    {
25        GL.invertCulling = true;
26    }
27
28    void OnPostRender()
29    {
30        GL.invertCulling = false;
31    }
32 }
33 }
```

### 3.4 三層スクリーンのスクリーン間移動について

文責：海木 一佳

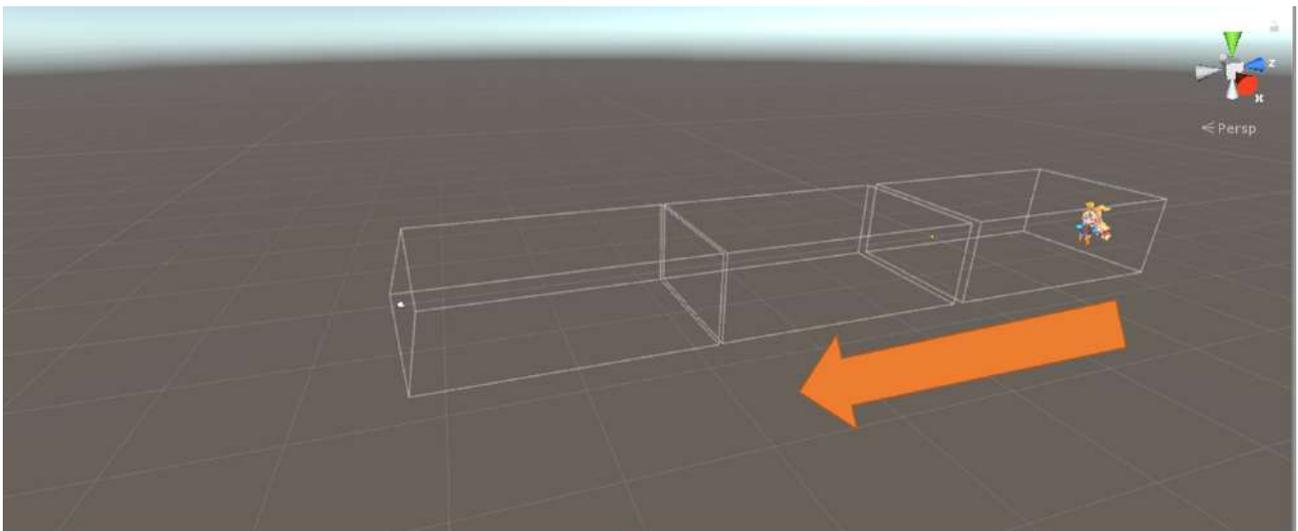
#### 3.4.1 スクリーン間移動の違和感

本プロジェクトは三層のスクリーンによって奥行きが再現されている。キャラクターが奥から近づいてくる、という映像表現を行う際にはキャラクターをスクリーン間を移動させる必要がある。(例：二層で踊っていたキャラクターが一層まで近づいてプレイヤーに挨拶を行う)。しかし、層と層の間には物理的な距離があり、純粋に移動させただけでは遠近感によって違和感が生まれる。特にユーザーにキャラクターが瞬間移動したように見えてしまう。

#### 3.4.2 移動方法考察

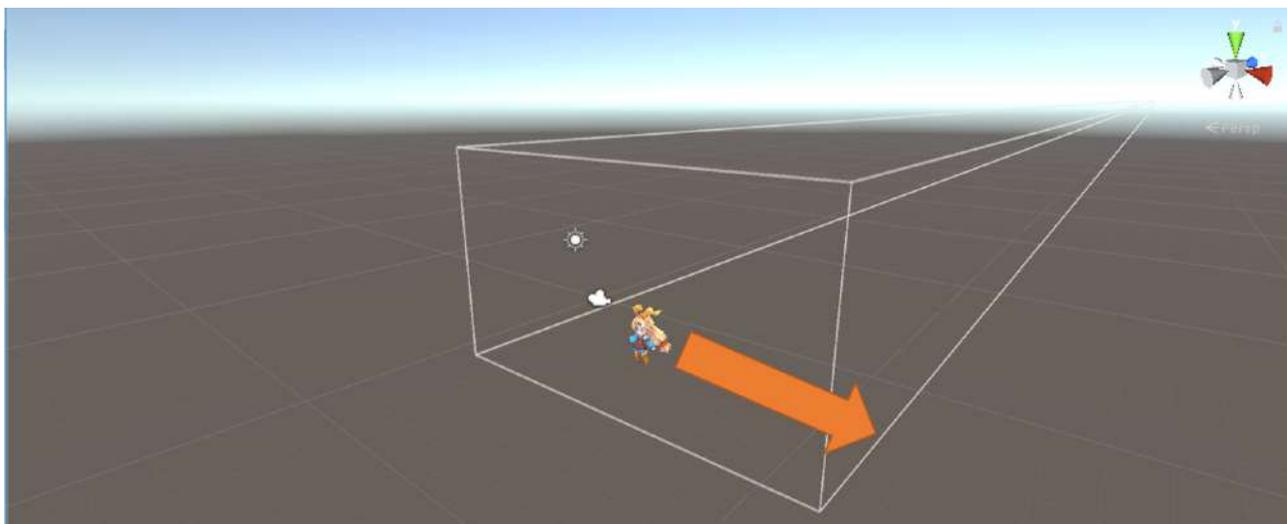
- 純粋な前進

図1のようにカメラを3つ並べ、キャラクターを直進させるとする。各カメラは各層の映像を撮っており、直進することでカメラの撮影領域から外れ、自動的に層の移動も行われていく。この方法は最も簡単だが、層の移動方法としては適切ではない。まず、キャラクターが近づいてきているように見えない。正投影で映像を撮っているため、近づけても遠近感は適用されず、近づいているように見えないのである。また、層の移動時にスクリーン間の物理的な距離によって瞬間移動しているように見える問題は解決しない。加えて、カメラから見切れる際に、キャラクターの体の中が透けて違和感を発生させる。



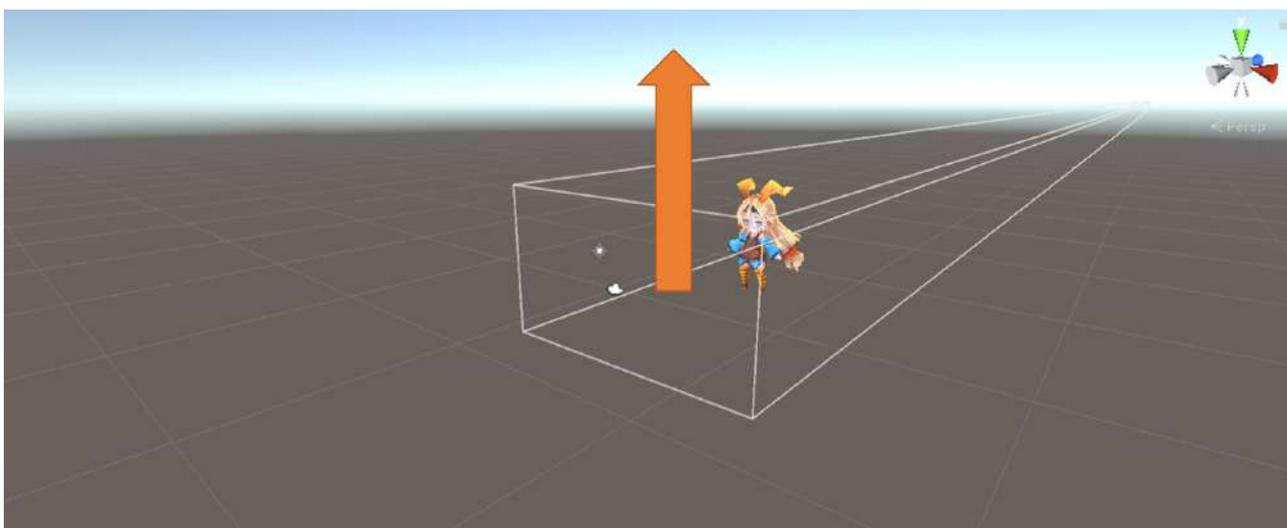
- 横に見切れさせる

図2のようにカメラを同じところへ3つ並べ、キャラクターを左右へ移動させ、カメラから左右に見切れた際に層の移動を行う。各カメラの撮影する要素はレイヤーで管理しており、層の移動を行う際は、その移動対称のオブジェクトのレイヤーを変更している。キャラクターには前進をさせず、少しずつ大きさを拡大していくことで近づいているかのように見せる。見切れている間にも拡大処理も行い、見切れている間にも近づいているようにする。この手法は層の移動に違和感をほとんど発生させなかったため、一度見切れさせる手法は非常に相性が良いと考えられる。



- 縦に見切れさせる

図2と同じようにカメラ、キャラクターを設置する。図2と異なるところは左右に移動させるのではなく、ジャンプして上方向で見切れさせ、降りてくる際に層の移動を行うことである。イメージとしてはよくバラエティ番組などでジャンプして目的地に到着させる映像表現である。こちらの手法も違和感なく見る事ができた。



### 3.5 三層スクリーンを利用したポケモン風 UI

文責：海木 一佳

#### 3.5.1 内容

ポケモンは選択肢、自分のポケモン、相手のポケモンという三層から UI が形成されている。実際に本プロジェクトの機材を利用し、三層を使って再現してみた。手前の層を選択肢、中層を自分のポケモン、奥の層を相手のポケモンにした。相性は非常に良く、奥行きを感じることができた。



#### 3.5.2 お借りしたもの

- Pu 様:Tda ミヅキ  
(<http://seiga.nicovideo.jp/seiga/im6655754>)
- Pikadude31451 様:ガオガエン  
(<http://seiga.nicovideo.jp/seiga/im6554729>)
- IVatu 様:フーディン  
(<http://seiga.nicovideo.jp/seiga/im6325239>)

## 4 活動で得られたもの

文責：程 瑞希

ハードの開発とソフトの開発で班員の担当を分けることで、各班員が各々の得意分野で開発することに専念できた。また、班員で集まってアイデア出しを行うことで様々な方向からのアイデアが出た。

## 5 展望

文責：程 瑞希

本プロジェクトでは、さまざまなアイデアが出た。以下にそのアイデアの例を記す。来年度の新歓や学園祭に向けてより様々な展示ができると思われる。

- メディアアートでは、床の部分に別の液晶画面を置くことでステージなどをより立体的に演出できると思われる。また、手前のスクリーンに譜面を配置したリズムゲーム制作がアイデアとして出た。実装には至らなかったが、三層であることによって譜面が映像と重ならずリズムゲームとしてとてもプレイしやすくなると思われる。
- ゲームに関して今回はポケモン風の UI の開発に終わったが、他にも横スクロールの平面アクションゲームや縦スクロールのシューティングゲームなど、ゲーム方面で様々な新しい UI を制作できると思われる。