

Réalité augmentée et interactions hybrides

Gaïa Thomas

ENSIM, Le Mans Université
1 rue Aristote, Le Mans, France

`gaia.thomas.etu@univ-lemans.fr`

Résumé Dans le cadre de ce projet, j’ai développé un jeu de réalité augmentée utilisant Unity et AR Foundation. Mon application crée une expérience où l’utilisateur contrôle une mascotte virtuelle qui réagit aux marqueurs thermiques dans l’environnement réel. Le serpent grandit au contact de sources chaudes et rétrécit près de sources froides, ce qui crée un gameplay basé sur l’exploration thermique de l’espace physique.

Keywords: Réalité Augmentée · Unity · AR Foundation · Détection d’images · Jeu interactif.

1 Introduction

1.1 Projet

Mon idée pour ce projet était de réaliser une mascotte qui permet à l’utilisateur d’essayer de réchauffer la pièce dans laquelle il se trouve. Lorsqu’il rencontre un point (marqueur) chaud, il grandit, et inversement avec un point froid.

L’objectif que j’ai fixé est simple : le joueur gagne la manche si le serpent atteint une certaine taille et ne disparaît pas à la fin de l’exploration de la pièce. J’ai voulu créer une expérience où l’utilisateur peut interagir avec sa mascotte en la faisant apparaître et surtout en la dirigeant dans l’espace réel.

J’ai décidé de réaliser ce projet seul pour me concentrer pleinement sur l’apprentissage des technologies de réalité augmentée et maîtriser tous les aspects du développement.

2 Motivation

2.1 Pourquoi un tel système ?

J’ai choisi de développer ce système pour plusieurs raisons.

Tout d’abord, je voulais explorer les possibilités offertes par Unity dans la cadre d’un projet de réalité augmenté. J’ai ainsi dû utiliser des technologies avec lesquelles je n’étais pas totalement à l’aise. Ce projet m’a semblé être un excellent moyen d’apprendre ces outils tout en créant quelque chose d’original et d’amusant.

L'idée du serpent thermique m'est venue en réfléchissant à un problème rencontré à l'école et lié à la gestion de la température. L'infrastructure nous fait vivre des étés très chauds et des hivers glacials. Je trouvais donc plutôt amusant et intéressant de montrer la différence de gain et de perte de chaleur dans une pièce de l'école.

3 Description du système

3.1 Aspect technique

Pour développer ce projet, j'ai utilisé Unity 2022.3.3 LTS avec le framework AR Foundation 5.1. J'ai suivi le tutoriel disponible sur GitHub (<https://github.com/Unity-Technologies/arfoundation-samples/tree/5.1>) pour maîtriser les bases.

Architecture J'ai organisé mon code autour de deux scripts principaux :

DragonGameManager : C'est le gestionnaire principal que j'ai créé. Il s'occupe de détecter les marqueurs thermiques grâce à l'ARTrackedImageManager, de placer le serpent dans l'espace avec l'ARRaycastManager, et de coordonner toutes les interactions.

DragonController : Ce script gère le comportement de mon serpent. J'y ai intégré la logique de température, les changements de taille, le système de refroidissement ambiant, et les conditions de victoire/défaite.

Système de détection J'ai créé une bibliothèque d'images de référence contenant 4 marqueurs différents :

- **Marqueurs chauds** : feu, radiateur
 - **Marqueurs froids** : flocon, fenêtre
- Chaque marqueur a une valeur thermique que j'ai définie :
- Feu/Radiateur : $+20^{\circ}\text{C}$
 - Flocon/Fenêtre : -15°C

La détection fonctionne en temps réel. Quand un marqueur est reconnu, mon système envoie l'information au serpent qui ajuste sa température en conséquence.

Mécanique de jeu que j'ai développée J'ai développé une mécanique de jeu avec un serpent dont plusieurs paramètres évoluent en temps réel. Le serpent possède une température variant de -10°C à $+80^{\circ}\text{C}$. Cela influence directement sa taille qui peut aller de 0.3 à 2.0. Un refroidissement ambiant constant de -1°C par seconde est appliqué, ce qui oblige le joueur à interagir efficacement dans un temps limité de 90 secondes. Trois conditions de fin de partie ont été programmées : une victoire est atteinte si la taille du serpent atteint au moins 1.8, tandis qu'une défaite survient si sa taille chute à 0.3 (le serpent disparaît) ou si le temps imparti s'écoule.

Interface utilisateur que j'ai créée Pour accompagner cette mécanique, j'ai conçu une interface utilisateur simple affichant en permanence la température actuelle du serpent, sa taille courante ainsi que le temps restant. Le positionnement du serpent se fait par une simple tap sur l'écran, grâce à l'intégration du système de détection de plans fourni par AR Foundation.

Défis techniques J'ai eu de nombreux défis pendant l'avancer de mon jeu et je n'ai pas pu tous les relever. Le premier étant que je ne suis pas parvenu à résoudre. Je ne parvenais pas à lier mon JDK à Unity. J'ai donc pu développer uniquement sur l'ordinateur de l'école. Lorsque j'ai développé mon projet l'image tracking fonctionnait et je pouvais bien faire apparaître mon serpent en touchant l'écran. Mais ne parvenant pas à le déplacer je n'ai pas pu tester l'image tracking.

4 Discussion et conclusion

4.1 Qu'avons-nous réalisé ? Points forts, points faibles

Ce que j'ai accompli Points forts : J'ai réussi à intégrer certaines fonctionnalités de base de la réalité augmentée dans Unity.

Points faibles :

Mon jeu n'est pas fonctionnel. Le serpent ne se déplace pas, ce qui m'a empêché de tester les différents marqueurs. Visuellement, le serpent reste très simple : je n'ai pas eu le temps d'intégrer des modèles 3D plus élaborés. À l'origine, je comptais créer un modèle sous Blender, mais j'ai dû me résoudre à utiliser des prefabs de l'Asset Store. Enfin, la détection des marqueurs dépend fortement des conditions d'éclairage : dans un environnement trop sombre, les marqueurs ne sont pas reconnus.

4.2 Quelles seraient les étapes suivantes si on voulait améliorer et/ou étendre ce projet ?

Principales : Réussir à faire fonctionner mon jeu.

Améliorations visuelles : J'aimerais remplacer mes objets primitifs par un vrai modèle 3D de serpent avec des animations. J'ajouterais des effets de particules pour le feu et la glace, et des sons pour rendre l'expérience plus immersive.

Jeu : Je pourrais ajouter différents niveaux de difficulté, un système de score, ou même plusieurs serpents avec des caractéristiques différentes.

5 Conclusion personnelle

Ce projet m'a énormément appris. Au-delà des aspects techniques (Unity, AR Foundation, programmation mobile), j'ai été confronté aux nombreux défis spécifiques au développement en réalité augmentée : optimisation des performances, création d'interactions intuitives, dépendance à l'environnement réel, etc.

J'aurais aimé disposer de plus de temps pour mener à bien ce projet et atteindre mes objectifs initiaux.