
Option: Système Intelligents & Multimédia

Module: Réalité Virtuelle

Rapport de projet sur la simulation de motos dans une ville virtuelle

Présenté par :

BIABA KUYA Jirince

MASOBELE MVITA Charly

TOUMBA NGONGO Christine

Encadreur: Dr Thong DANG

Date de soumission: 05-Mai-2025

Contents

Chapter 1: Simulation de motos dans une ville virtuelle	3
1.1 Introduction	3
1.1.1 Présentation du sujet	3
1.1.2 Objectifs du projet	3
1.1.3 Contraintes	3
1.2 Processus de développement	4
1.2.1 Étapes clefs	4
1.2.2 Détails de chaque étape	4
1.3 Choix techniques	5
1.4 Difficultés rencontrées	5
1.5 Solutions mises en œuvre	6
1.6 Résultat final	6
1.7 Pistes d'amélioration	7
1.8 Conclusion	9
1.8.1 Lien vers le code source	9

Chapitre 1: Simulation de motos dans une ville virtuelle

1.1 Introduction

1.1.1 Présentation du sujet

Dans le cadre de notre projet de simulation 3D, nous avons choisi de concevoir et de réaliser une simulation réaliste de motos évoluant dans une ville virtuelle. Ce sujet nous a particulièrement motivés, car il combine des aspects techniques variés tels que la modélisation, l'animation, la programmation interactive et la gestion d'un environnement urbain vivant.

Ainsi nous avons mis en pratique nos compétences en modélisation, animation, développement et gestion collaborative.

Tout au long de cette aventure, nous avons travaillé de manière étroite et complémentaire, partageant les tâches et prenant ensemble les décisions essentielles. Ce rapport présente notre démarche, nos choix techniques, les difficultés rencontrées ainsi que les solutions que nous avons mises en œuvre pour mener à bien ce projet.

1.1.2 Objectifs du projet

L'objectif principal de notre projet était de simuler de manière réaliste le déplacement de motos dans un environnement urbain virtuel. Pour cela, nous avons cherché à :

- Reproduire fidèlement le comportement des motos : accélération, freinage, virage et arrêt.
- Synchroniser les mouvements des motards avec ceux des motos pour un rendu naturel.
- Permettre à un utilisateur de prendre le contrôle d'une moto en vue à la troisième personne.
- Animer dynamiquement la ville en générant automatiquement des motards pour rendre l'environnement vivant.

1.1.3 Contraintes

- Temps limité à 4 semaines.

- Développement en parallèle de l'apprentissage d'Unity et des outils associés.
- Les ressources 3D utilisées (ville, personnages, moto) sont lourdes et nécessitent une bonne gestion.
- Le projet devait intégrer des personnages animés sans impacter les performances.
- Les animations et les collisions devaient rester crédibles et réalistes.
- Assurer un équilibre optimal entre performance et réalisme.

1.2 Processus de développement

1.2.1 Étapes clefs

Semaine	Tâches principales
S1	Initialisation du projet, intégration de la ville 3D
S2	Implémentation du contrôle de la moto, caméra dynamique
S3	Ajout de la population, animations, spawn aléatoire
S4	Finalisation, nettoyage du projet, mise en ligne sur GitHub

1.2.2 Détails de chaque étape

1. Semaine 1 : Intégration de la ville

- Importation d'un modèle .fbx représentant une ville.
- Réglages de la scène Unity : positionnement, lumière, skybox.
- Ajout de MeshCollider sur les bâtiments et routes pour éviter que la moto ou les personnages traversent le décor.

2. Semaine 2 : Contrôle de la moto

- Intégration du modèle 3D de la moto.
- Ajout d'un contrôleur (script C# + Rigidbody).
- Création d'un système de caméra dynamique avec Cinemachine pour suivre la moto en temps réel.

3. Semaine 3 : Animation de la population

- Création de prefabs de personnages animés (Idle, Walk, Run).
- Script PeopleController pour gérer les animations aléatoires.

- Script PopulationSpawner pour faire apparaître dynamiquement des humains dans la ville.
- Réduction de la zone de spawn pour éviter qu'ils apparaissent sur les routes.

4. Semaine 4 : Optimisation et déploiement

- Nettoyage de la scène et ajustement des positions.
- Correction des bugs (navigation etc ...).
- Enrichissement de l'environnement (ajout de détails urbains).
- Ajout d'un .gitignore pour Unity.
- Intégration de Git LFS pour gérer les fichiers .fbx trop lourds.
- Envoi complet du projet sur GitHub pour permettre le travail collaboratif.
- Préparation des supports de communication et de la présentation finale.

1.3 Choix techniques

- Unity : Moteur de développement permettant la création d'environnements interactifs en 3D. Il a été choisi pour sa prise en main rapide et sa puissance graphique.
- Blender : Outil de modélisation et d'animation 3D pour la création des personnages et des éléments du décor.
- GIMP : Logiciel de retouche graphique pour l'élaboration de textures, icônes et éléments visuels.
- GitHub / GitLab : Plateformes de gestion de version pour faciliter le travail collaboratif, le suivi des modifications et la gestion du code source.
- Git LFS a permis d'éviter le rejet des gros fichiers .fbx par GitHub.
- Cinemachine permet un contrôle fluide de la caméra sans codage complexe.
- Animation via Animator Controller pour rendre les personnages vivants.
- Prefabisation des éléments récurrents (humains, moto) pour simplifier le réemploi.

1.4 Difficultés rencontrées

- Prise en main de l'interface de Unity: difficultés à maîtriser l'outil Unity et les différents outils et options,
- Compréhension des scripts: le C# est nouveau langage donc difficile de le comprendre correctement et de pouvoir le modifier,

- Les personnages et les motos tombaient dans le vide ou traversaient les murs.
- La caméra ne suivait pas toujours correctement la moto.
- Synchronisation des modules : Intégrer les animations et les scripts de contrôle ensemble sans conflits a demandé des ajustements.
- Les personnages n'apparaissaient pas dans la vue Game (problème de layer).

1.5 Solutions mises en œuvre

- Utilisation de Git LFS pour les fichiers .fbx > 100 Mo.
- Ajout de MeshCollider sur la ville pour gérer les collisions.
- Utilisation de Cinemachine Virtual Camera bien fixée à la moto.
- Réglage du Culling Mask et du plan de clipping de la caméra.

1.6 Résultat final

Une ville virtuelle en 3D fonctionnelle, avec textures et collisions réalistes.



Figure 1.1: Enter Caption

Une moto entièrement contrôlable (accélération, freinage, virage).

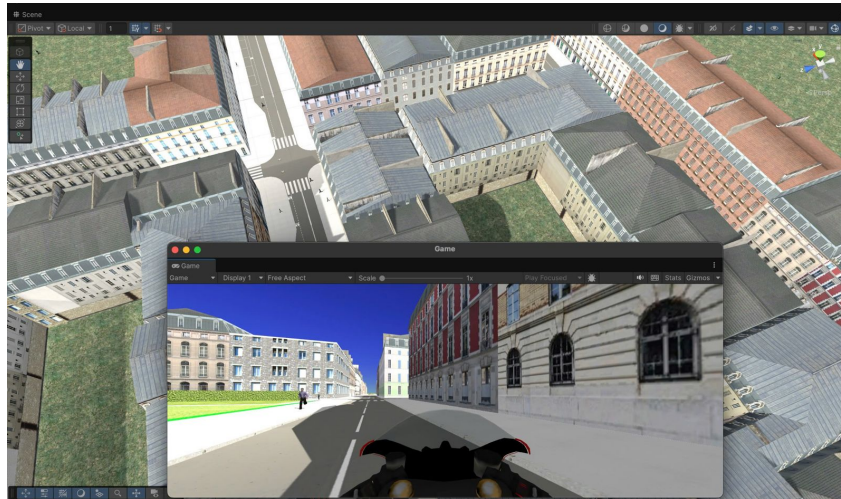


Figure 1.4: Enter Caption



Figure 1.2: Enter Caption

Caméra fluide et dynamique, fixée à l'arrière de la moto.



Figure 1.3: Enter Caption

Une population animée (Idle/Walk/Run), répartie automatiquement dans les zones piétonnes.

1.7 Pistes d'amélioration

- Ajout d'autres motos autonomes (IA circulant en ville)

- Système de feux de signalisation avec détection
- Intégration de sons (moteur, bruit de ville)
- Interface utilisateur : carte, vitesse, bouton start/reset
- Démo vidéo et interface de menu de lancement
- Animation dynamique des piétons (marcher automatiquement)
- Ajout de feux tricolores et circulation routière
- Collisions moto/piétons

1.8 Conclusion

Ce projet a permis de mettre en pratique la création d'un environnement interactif 3D réaliste avec Unity. Grâce à l'intégration de la moto, de la population animée et d'une ville cohérente, l'ensemble forme une simulation crédible. Les outils comme Git, GitHub et Git LFS ont renforcé les bonnes pratiques de gestion de projet logiciel, en particulier pour la collaboration. Le projet peut évoluer vers une version multijoueur ou une ville encore plus interactive avec des feux de circulation, des PNJ (Personnage Non Joueur, un personnage contrôlé par l'ordinateur, pas par le joueur) motorisés et une interface utilisateur plus riche.

1.8.1 Lien vers le code source

[Github](#)