

Rapport VisuTerrImac

Julien Calabrese - Léo Genot - Elise Gondange

Présentation du projet

Dans le cadre de nos cours de synthèses d'images et de programmation algorithmique du S2, il nous a été demandé de réaliser un projet de modélisation 3D. Les technologies et langages utilisés pour ce projet sont le C++ et l'OpenGL.

Notre objectif était de créer un terrain à partir d'une HeightMap dans lequel l'utilisateur peut se déplacer. Ce terrain devra être texturé, en relief, et contenir des billboards qui permettent d'afficher des arbres. Les textures et l'affichage se font en OpenGL, tandis que l'optimisation des ressources se fait grâce à un quadtree en C++.

Présentation de l'équipe

Notre équipe est un trio, composé de Julien Calabrese, Léo Genot et Elise Gondange. Nous avons tous les trois découvert les bases de l'OpenGL ce semestre, ce projet était donc notre premier réel exercice de taille. De plus, aucun de nous ne vient de prépa, certaines notions mathématiques, notamment liées aux vecteurs et aux matrices, nous ont été compliquées à comprendre.

Organisation de travail

Méthode de travail

On a souhaité découper le projet en différentes petites tâches afin d'être plus efficace dans notre travail et d'apprendre plus facilement les subtilités de l'OpenGL. Pour cela, nous sommes tout d'abord partis sur l'aspect "visuel" du projet qui nous semblait être le plus facile pour commencer.

Nous nous sommes répartis ces différentes tâches : Elise s'est occupée de créer la skybox, Léo d'intégrer la caméra et Julien de créer le terrain. Lorsque l'un de nous avait une difficulté nous n'hésitions pas à nous contacter depuis Discord.

Par la suite, nous nous sommes rapidement rendu compte que sans la création du quadtree, beaucoup trop de ressources étaient requises et il nous était impossible de continuer à travailler sur le projet. Nous nous sommes donc attaqués au quadtree, qui nous a pris la majeure partie du temps. Et grâce à des ressources telles que le site vterrain.org nous avons réussi à intégrer le quadtree ainsi que le LOD au terrain (en modifiant un peu notre lecture de heightmap).

Une fois avoir notre terrain généré par le quadtree, nous avons travaillé sur les fonctionnalités supplémentaires à intégrer comme les billboards et les textures.

Fonctionnalité technique

Une des fonctionnalités que nous avons implémentées est le ray casting. Grâce à cela, nous avons pu permettre à l'utilisateur d'augmenter la hauteur d'un point du terrain en le visant et en cliquant. Techniquement, cela revient à modifier la valeur du triangle du terrain qui est visé. Le ray tracing pourrait aussi nous permettre dans le futur de créer un éclairage dynamique ou d'ajouter des objets sur le terrain.

Difficultés rencontrées

Nous avons peu pratiqué la 3D en OpenGL pendant les sessions de TD de synthèse d'image, ce projet était donc un peu une découverte pour nous.

La plus grosse difficulté a été d'implémenter le quadtree au projet OpenGL. C'est la partie sur laquelle nous avons passé le plus de temps et suite à laquelle nous n'avons pas eu le temps d'implémenter toutes les fonctionnalités du sujet.

Aspects qui diffèrent du sujet initial

Nous n'avons pas pu implémenter l'éclairage, le soleil, le frustum culling et la réparation des cracks au projet.

Pour ce qui est de la heightmap au début nous utilisions un fichier BMP et nous lisions chaque pixel pour régler la hauteur de notre terrain. Mais dans notre dernière version nous avons utilisé la fonction de lecture de fichier intégré au c++ qui stocke directement tout le contenu "Brut" du fichier dans un vecteur.

La caméra elle aussi diffère un peu du sujet, en effet nous avons créé notre propre système de "vision" grâce à la trigonométrie, permettant notamment d'avoir ce rendu type "FPS".

Nous avons aussi ajouté une fonctionnalité pour augmenter la hauteur d'un point sur le terrain à l'aide du raycasting.

Conclusion / Améliorations souhaitées

Finalement, ce projet nous a tous permis de découvrir de manière un peu plus concrète l'OpenGL.

Si nous pouvions améliorer ou changer la façon dont nous avons travaillé, nous aurions commencé par créer le quadtree pour ensuite s'attaquer à la partie "visuel". Cela nous aurait sûrement permis de gagner du temps car en implémentant le quadtree plus tard, nous avons dû revenir sur des parties faites auparavant.

De plus, nous aurions apprécié avoir plus de temps sur ce projet afin de réussir à implémenter toutes les fonctionnalités demandées qui étaient conséquentes pour des débutants.