

Représentation du réel par génération procédurale

La synthèse d'images numériques est un domaine qui nous intéresse, car elle possède de nombreuses applications dans la vie de tous les jours, par exemples dans le cinéma ou dans les jeux vidéos.

Nous nous sommes ainsi dirigés vers la génération procédurale et ses différents algorithmes, puis nous nous sommes penchés sur le bruit de Perlin en particulier car cet algorithme exploite une méthode d'optimisation abordable à comprendre et adaptée au langage mis à notre disposition (Python).

Ce TIPE fait l'objet d'un travail de groupe.

Liste des membres du groupe :

- *CHAPUIS Martin*
- *SCHIEB Antoine*

Positionnement thématique (étape 1)

INFORMATIQUE (Informatique pratique), INFORMATIQUE (Informatique Théorique), MATHEMATIQUES (Autres).

Mots-clés (étape 1)

Mots-Clés (en français)	Mots-Clés (en anglais)
<i>Aléatoire</i>	<i>Random</i>
<i>Génération procédurale</i>	<i>Procedural generation</i>
<i>Bruit de Perlin</i>	<i>Perlin noise</i>
<i>Texture</i>	<i>Texture</i>
<i>Carte de hauteur</i>	<i>Heightmap</i>

Bibliographie commentée

Dès les débuts de l'informatique, on a cherché à représenter le réel grâce au numérique (cinéma, jeux vidéo). Cette représentation porte sur différents domaines tels que l'image (paysage complet en fond, herbe, ombre), le son (l'eau d'une rivière), etc. Nous nous intéressons plus particulièrement à la représentation graphique du réel. On trouve différentes techniques se basant sur le contrôle de l'aléatoire qui joue un rôle important dans cette représentation. Ce contrôle s'effectue par l'intermédiaire de la génération procédurale [1]. Celle-ci consiste à créer du contenu numérique à partir d'un algorithme faisant intervenir de l'aléatoire, appelé "Algorithme de génération procédurale" afin d'obtenir des images réalistes, autant sur le relief que sur les couleurs, notamment dans les jeux vidéo [2].

Il existe différentes techniques de génération procédurale, par exemple l'algorithme des Diamants-Carrés ou le Bruit de Perlin. L'algorithme des Diamants-Carrés s'apparente au procédé de génération de fractales. En effet, il consiste à appliquer le même algorithme sur des espaces de plus en plus petits. Dans le cadre de la génération de carte, cet espace est représenté par un tableau de

valeurs qui caractérisent l'altitude du point correspondant. Cet algorithme définit donc aléatoirement quatre valeurs correspondant aux coins du tableau puis détermine les altitudes des autres points par moyenne (d'abord celui au centre en ajoutant, à cette moyenne, un décalage aléatoire puis ensuite les milieux des quatre segments formés par les quatre coins) et enfin réitère le processus sur les nouveaux diamants ou carrés délimités par les nouveaux points [3]. Le Bruit de Perlin, appliqué également sur un tableau, repose sur l'utilisation de vecteurs. En effet, ce bruit va découper le tableau de valeurs en différentes zones de même taille et attribuer à chaque zone quatre vecteurs unitaires générés aléatoirement aux coins [4]. Il affecte ensuite, à chaque point d'une même zone, une valeur qui correspondra à son altitude ou à sa couleur, obtenue par un calcul de produits scalaires faisant intervenir sa position et les vecteurs générés aux quatre coins, suivi d'une interpolation.

Il existe différentes manières de représenter ces valeurs, par exemple sous forme de carte [5] en faisant correspondre la valeur de chaque point à une couleur ou encore directement en 3D [6] en affectant une altitude à la valeur de chaque point et enfin on peut mélanger ces deux représentations : cette dernière méthode est utilisée pour générer des heightmaps qui sont des cartes utilisées pour les représentations des paysages dans les images numériques, dans les jeux vidéo par exemple [7]. La diversité de représentation des valeurs et du contrôle de celles-ci, par modification de la procédure selon l'objectif de l'algorithme, offre de nombreuses applications. Parmi elles, on trouve la génération de textures. Une texture est une image 2D appliquée sur un objet 3D ou 2D afin de lui donner l'aspect voulu [8]. Il existe deux types de textures, celles créées par l'homme, directement à partir de photos ou d'images, et celles générées de manière procédurale à l'aide d'algorithmes, ces dernières ayant l'avantage d'être moins coûteuses en mémoire, et généralement plus flexibles dans leur utilisation. Ainsi, les algorithmes de génération procédurale constituent un outil pour créer des textures qui répondent aux critères souhaités.

Problématique retenue

Comment écrire un algorithme de génération procédurale du réel capable d'utiliser l'aléatoire tout en restant représentatif du réel, afin que l'utilisateur puisse en contrôler les paramètres (taille, échelle, couleurs, résolution) et représenter ces bruits, ayant des applications dans plusieurs domaines, de façon adaptée aux demandes ?

Objectifs du TIPE

Créer un algorithme de génération procédurale en utilisant Python, avec pour objectifs :

- I) La conception du bruit ;
- II) Les améliorations de celui-ci ;
- III) La représentation des valeurs générées et application à des cas pratiques

Mes objectifs personnels sont pour chaque point :

- I) coder cette procédure sur Python et de l'adapter à l'utilisation des tableaux ;

- II) améliorer et l'optimiser l'algorithme à l'aide de la création d'une fonction poids ;
- III) représenter ces valeurs en 3D.

Abstract

Our work consisted of generating random noises which represented different aspects of the real world. We had to focus on optimizing them and representing them. Starting off with already existing noise algorithms, such as the Perlin Noise, we developed our own program in two and three dimensions. We tried different ways of displaying our noises following user-related criteria, namely grayscale, color, geographic maps, and landforms. We were able to process real images to directly apply our noise as a synthetic texture.

Références bibliographiques

- [1] FR.WIKIPEDIA.ORG : Génération procédurale :
https://fr.wikipedia.org/wiki/G%C3%A9n%C3%A9ration_proc%C3%A9durale
- [2] INDIUS.ORG : La génération procédurale dans le jeu vidéo :
<http://indius.org/dossiers/generation-procedurale-jeu-video/>
- [3] GEEKLABORATORIES.COM : Algorithme Diamants-Carrés: Génération de heightmaps :
<http://geeklaboratories.com/789-algorithme-diamants-carres-generation-de-heightmaps>
- [4] OPENCLASSROOMS.COM : Bruit de Perlin : <https://openclassrooms.com/courses/bruit-de-perlin>
- [5] WWW.GRAPPA.UNIV-LILLE3.FR : Pixel: introduction : http://www.grappa.univ-lille3.fr/~coulom/Python/fonctions_prog_pixel.html
- [6] MATPLOTLIB.ORG : mplot3d tutorial : http://matplotlib.org/mpl_toolkits/mplot3d/tutorial.html
- [7] LINUXFR.ORG : Je crée mon jeu vidéo E10: génération procédurale de carte (partie 1) :
<http://linuxfr.org/news/je-cree-mon-jeu-video-e10-generation-procedurale-de-carte-partie-1>
- [8] WIKI.BLENDER.ORG : Textures procédurales, Blender :
<https://wiki.blender.org/index.php/Doc:FR/2.4/Manual/Textures/Types/Procedural>