

224 - DataCube Raycasting

Antoine LESTRADE

Travail de bachelor 2021

Informatique - Développement Logiciel et Multimédia

Professeur : Stéphane Gobron

Expert : Antoine Widmer

Description

De nombreux chercheurs doivent traiter et travailler avec des données provenant de l'observation terrestre (EO) par satellite. L'EO couvre divers champs d'applications comme l'agriculture, l'urbanisme ou l'étude du climat. Les données issues de l'EO sont représentées sous formes de matrices de taille variable aussi appelées *datacube*.

Le projet consiste à visualiser des *datacube* en temps réels dans un canevas HTML à l'aide de WebGL. Il utilise un algorithme de *raycasting* accéléré par GPU spécialement conçu pour le rendu volumique de ces *datacube*. L'objectif principal est de pouvoir proposer un outil de visualisation simple et performant pour visualiser ces volumes de données.

Parmi les objectifs à atteindre à la fin de ce projet :

- Rendu volumique et de surface de *datacube* ;
- Rendu des dérivés par rapport à une valeur seuil ;
- Support de *datacube* de taille 16^3 , 64^3 , 256^3 , 1024^3 ;
- Implémentation de l'algorithme dans un *notebook Jupyter* ;

Déroulement

Les phases du projet comprendront dans l'ordre :

- Comprendre la théorie du modèle de rendu *raycasting* ;
- Implémenter l'algorithme en WebGL pour le rendu du volume ;
- Permettre le rendu au sein d'un *notebook Jupyter* ;
- Tester le rendu pour des *datacube* de taille 256^3 ;
- Implémenter le rendu de surface avec seuillage ;
- Implémenter un seuillage pour visualiser une couche spécifique du *datacube* et ses différences (i.e. différentiel direct) avec les autres couches ;
- Implémenter une solution de *raycasting* supportant un *datacube* de taille 1024^3 .

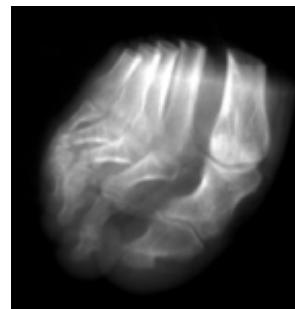
Perspectives

La majorité des objectifs fixés a été atteinte. Le programme met à disposition plusieurs modes de rendu, qui répondent chacun aux exigences demandées concernant la visualisation 3D de *datacube*. Des optimisations concernant le rendu et les textures ont été nécessaires pour parvenir à traiter des grands volumes de données avec le moins d'impact sur les performances. Pour la suite, ces rendus pourront être intégrés dans une boîte à outils en Python à destination de la communauté scientifique ou pourront être utilisés dans le cadre d'applications Web pour le grand public. Des optimisations sont encore nécessaires, notamment pour permettre le rendu de *datacube* 1024^3 sur des configurations matérielles plus modestes.

Résultats

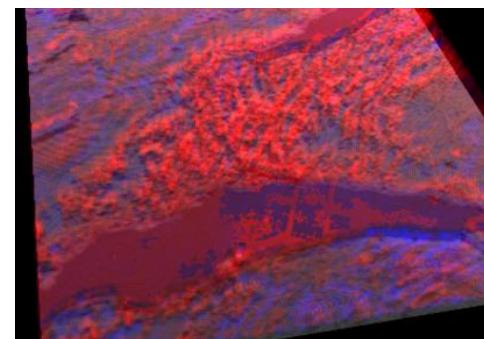
Le programme supporte des *datacube* de dimensions 16^3 , 64^3 et 256^3 . Les *datacube* 1024^3 sont supportés mais sous certaines conditions. Les modes de rendu présentés ci-dessous fonctionnent en temps réel dans un *notebook Jupyter*.

Deux modes de rendu sont disponibles pour les *datacube* avec trois dimensions spatiales : un rendu X-Ray et un rendu de simulation de surface implicite. Il est possible de modifier la précision de l'algorithme de *raycasting*, la valeur de seuil pour la surface et la palette de couleur utilisée



Exemple de rendu X-Ray à gauche et de rendu de surface à droite

Le rendu Dérivé est conçu pour les *datacube* avec deux dimensions spatiales et une dimension temporelle. Il permet de visualiser l'évolution d'une surface 2D dans le temps et de mettre en évidence les différences. Il est possible de modifier en temps réel l'intervalle à afficher et le temps t de référence.



Exemple de rendu Dérivé avec *datacube* issu de l'observation terrestre