

Accélération du ray-tracing avec des Bounding Volume Hierarchies

Se situant à l'intersection de nombreux domaines de l'informatique, allant du multithreading aux algorithmes gloutons, l'optimisation du ray-tracing est un problème au coeur de la recherche actuelle et dont les applications pratiques sont nombreuses. J'ai été séduit par la diversité des approches proposées par la communauté scientifique.

L'outil numérique est un allié très précieux des architectes et urbanistes d'aujourd'hui. Il permet notamment à ces derniers de visualiser et de présenter leur travail en amont avec des rendus toujours plus réalistes. À condition, bien sûr, de pouvoir générer ces images en un temps raisonnable.

Positionnement thématique (ÉTAPE 1) :

- *INFORMATIQUE (Informatique pratique)*

Mots-clés (ÉTAPE 1) :

Mots-clés (en français)

Lancer de rayon

Hiérarchies de boites englobantes

Heuristique de l'aire

Binning

Mots-clés (en anglais)

Ray-tracing

Bounding Volume Hierarchies

Surface Area Heuristic

Binning

Bibliographie commentée

Au cours de la dernière décennie, le *ray-tracing* est devenu une alternative viable à la *rasterization* pour le rendu d'image non seulement *offline* (comme pour les films), mais également *online* (comme dans les jeux vidéos). Le *ray-tracing* se veut plus réaliste : il s'agit de simuler le comportement physique des rayons de lumière dans une scène. Pour chaque pixel de l'image, on trace un rayon virtuel ayant la caméra pour origine et on cherche l'intersection la plus proche entre ce rayon et les *primitives* (ici, on se limitera aux triangles) qui composent la scène. [1] Cependant, trouver une telle intersection s'avère coûteux si l'on se contente de tester naïvement tous les triangles. Un pan de la recherche en la matière se consacre donc à l'étude de structures de données permettant d'accélérer la recherche de telles intersections, en éliminant des groupes entiers de triangles, à la manière d'une recherche dichotomique.

Nous nous intéresserons à une structure largement utilisée en pratique : les *BVHs (Bounding Volumes Hierarchies)*. Il s'agit d'une structure d'accélération qui implémente une stratégie de division ensembliste. L'objectif est de construire un arbre dont les feuilles forment une partition de l'ensemble des *primitives*. Chaque noeud interne correspond alors à une partie des *primitives*, dont il connaît la *AABB (axis-aligned bounding box)* [2]. Pour obtenir un résultat satisfaisant, il

est nécessaire de guider la construction de l'arbre par une heuristique. Nous utiliserons la célèbre *SAH* (*Surface Area Heuristic*) qui évalue géométriquement la probabilité qu'un rayon traversant l'*AABB* d'un noeud traverse également celle d'un de ses enfants.

En adoptant une méthode de construction *top-down*, on a, à chaque étape, $O(n)$ partitions raisonnables sur lesquelles évaluer la *SAH*, où n désigne le nombre de *primitives* d'un noeud. Cependant, ce procédé devient très coûteux dès lors que le nombre de triangles est important. Afin de limiter le nombre de partitions à tester, nous utiliserons une méthode de construction significativement plus rapide appelée *binning* [3] qui produit des arbres de qualité équivalente. L'idée est de diviser l'espace en un nombre fixé de *bins* délimitées par des plans équidistants puis de projeter chacune des *primitives* dans une *bin*. On partitionne alors selon le plan qui minimise la *SAH*. En pratique, un petit nombre de *bins* suffit.

En testant cette implémentation sur plusieurs scènes, on observe une importante dégradation des performances de rendu pour les scènes dont les triangles ont des tailles trop variées - une caractéristique commune à de nombreux *meshes* architecturaux. Elle s'explique par l'important chevauchement entre les *AABB* des noeuds. Pour y remédier, Stich M. et al. propose alors une approche hybride nommée *Spatial Splits* [4] qui consiste à évaluer simultanément le coût d'une division ensembliste et d'une division spatiale pour chaque noeud. Une primitive peut alors apparaître dans chacun des enfants d'un noeud, en échange d'un chevauchement nul des *AABB* de ceux-ci. On évaluera des plans selon une technique similaire au *binning*, appelée *chopped binning*.

Problématique retenue

Il s'agit de mesurer l'influence de la géométrie des scènes sur la qualité d'un *BVH* simple puis de proposer une méthode de construction plus sophistiquée sans détériorer le temps de construction de l'arbre.

Objectifs du TIPE du candidat

- Programmer un moteur de *ray-tracing* élémentaire implémentant un *BVH* en langage C.
- Mettre en évidence l'influence de l'homogénéité de la taille des triangles sur la performance du rendu à l'aide de *heatmaps*.
- Implémenter l'utilisation de *spatial splits* selon la méthode de *chopped binning* et comparer avec les résultats précédemment obtenus.
- Discuter les nombreux choix d'implémentation et optimisations possibles comme la parallélisation du processus de construction.

Références bibliographiques (ÉTAPE 1)

[1] PHARR M., JAKOB W., HUMPHREYS G. : Physically Based Rendering: From Theory to Implementation : <https://www.pbr-book.org/3ed-2018/contents>

[2] DANIEL MEISTER, SHINJI OGAKI, CARSTEN BENTHIN, MICHAEL J. DOYLE, MICHAEL GUTHE, JIŘÍ BITTNER : A Survey on Bounding Volume Hierarchies for Ray Tracing : https://meistdan.github.io/publications/bvh_star/paper.pdf

[3] WALD I. : On Fast Construction of SAH-based Bounding Volume Hierarchies :

<https://www.sci.utah.edu/~wald/Publications/2007/ParallelBVHBuild/fastbuild.pdf>

[4] TICH M., FRIEDRICH H., DIETRICH A. : Spatial Splits in Bounding Volume Hierarchies

: <https://www.nvidia.in/docs/IO/77714/sbvh.pdf>

DOT

[1] : *En septembre, étude du fonctionnement du ray-tracing et des structures d'accélération simples, puis implémentation d'un moteur de ray-tracing sans structure d'accélération*

[2] : *En novembre, implémentation d'un BVH simple sur le moteur, parallélisation de toutes les portions du moteur, recherche de scènes intéressantes pour les mesures à venir*

[3] : *Début janvier, rencontre avec Benjamin Segovia (responsable ray-tracing chez Sony) pour approfondir ma présentation et discuter de l'état de l'art*

[4] : *En janvier, étude des Spatial Splits et implémentation de ces-derniers sur le moteur*

[5] : *En février, réalisation des premières mesures, les scènes précédemment trouvées ne permettent pas de mettre suffisamment en avant l'impact des Spatial Splits*

[6] : *En mars, après avoir testé de nombreuses nouvelles scènes, sans malheureusement pouvoir trouver de scène "de ville" qui convienne, une scène en particulier sort du lot : le powerplant*