

Visualização e Iluminação II - Projectos

Luís Paulo Santos – Dep. de Informática

Universidade do Minho

Abril, 2018

Descrevem-se abaixo os projectos propostos para Visualização e Iluminação II, que incluirão também as 9 horas da UC Tecnologias e Aplicações alocadas a esta temática.

Os projectos estão classificados em 3 níveis de dificuldades: D1, D2 e D3. Os projectos D1 têm uma componente mínima de programação, sendo que uma avaliação positiva exige o cumprimento de uma grande percentagem dos objectivos propostos. Os projectos D3 exigem uma definição muito clara da arquitectura do *software* a desenvolver, sendo que uma avaliação positiva considerará como muito relevante a qualidade, maturidade e exequibilidade desta planificação. O nível D2 situa-se entre os dois extremos D1 <-> D3.

Os projectos devem ser desenvolvidos em grupos de 2 alunos. Como resultados finais a entregar incluem-se: uma apresentação (a realizar também oralmente em data a combinar), um relatório escrito e o produto do próprio projecto (código, modelos, etc.).

A lista de projectos abaixo não é de forma alguma limitativa. Um grupo de alunos com interesses dentro desta temática poderá propor e discutir com o docente o projecto que pretendem realizar.

Lista de projectos:

1. **Descrição do interface DirectX RayTracing e demo/código base** (dificuldade D2)
2. **Image based lighting com Embree: convergência e desempenho** (dificuldade D3)
3. **Elevado número de fontes de luz com Embree: convergência e desempenho** (dificuldade D1)
4. **Interacção Embree com motor de física** (ver https://en.wikipedia.org/wiki/Physics_engine e [https://en.wikipedia.org/wiki/Bullet_\(software\)](https://en.wikipedia.org/wiki/Bullet_(software))) (dificuldade D3)
5. **Embree e Bidirectional path tracing** (dificuldade D3)
6. **Melhor convergência usando geradores de números pseudo-aleatórios não correlacionados** (dificuldade D3)
7. **Modelação de um número de cenas** (pelo menos 5) demonstrando: (dificuldade D1)
 - indoor and outdoor;
 - número variado de primitivas geométricas e instanciação;
 - fenómenos difusos / especulares / *glossy* ;
 - número variado de fontes de luz;
 - configurações difíceis: *teapot in stadium, light through the key hole*.
8. **Optimização do desempenho recorrendo ao ISPC** (dificuldade D2)
9. **Optimização do desempenho recorrendo ao *streaming* e/ou aos pacotes de raios** (dificuldade D2)
10. **Optimização do desempenho usando múltiplas máquinas** (ex., com MPI no *search*) (dificuldade D3)

Cronograma

A entrega de todos os resultados finais e respectiva apresentação deverão acontecer até ao final de Junho.

Breve descrição dos projectos

1. **Descrição do interface DirectX RayTracing e demo/código base** (dificuldade D2)

A Microsoft disponibilizou há poucas semanas o DirectX RayTracing. Pretende-se que os alunos apresentem uma descrição das características essenciais desta API no formato de um tutorial acompanhado de uma demo. Este conjunto de materiais (apresentação, demo e tutorial) deverão constituir-se como um ponto de partida interessante para futuros alunos.

2. **Image based lighting com Embree: convergência e desempenho** (dificuldade D3)

Image based lighting consiste em utilizar mapas de radiância (imagens HDR) como fontes de luz. Normalmente estas imagens são mapeadas no interior de uma esfera virtual que contém toda a cena. *shadow rays* são disparados em determinadas direcções (estocasticamente seleccionadas, se se usar Monte Carlo) para aferir da visibilidade e fluxo radiante desta fonte de luz nessa direcção. São também designados por *environment map*.

Os tutoriais do Embree não suportam estas fontes de luz. Pretende-se desenvolver uma classe que as suporte, com especial ênfase na funcionalidade correcta. Isto é, aspectos como a taxa de convergência melhorada devida a amostragem baseada em importância extremamente sofisticadas e dependentes do do próprio *environment map*, não serão exigidas.

3. **Elevado número de fontes de luz com Embree: convergência e desempenho** (dificuldade D1)

Numa cena com um número elevado de fontes de luz o tempo dedicado ao processamento dos *shadow rays* rapidamente se transforma no custo mais significativo do *renderer*. Pretende-se comparar alguns métodos baseados em integração de Monte Carlo que seleccionam de forma inteligente qual a fonte de luz a amostrar (por exemplo, considerando as potências, áreas e distâncias relativas das várias fontes de luz).

4. **Interação Embree com motor de física** (ver https://en.wikipedia.org/wiki/Physics_engine e [https://en.wikipedia.org/wiki/Bullet_\(software\)](https://en.wikipedia.org/wiki/Bullet_(software))) (dificuldade D3)

Demonstrar o desenvolvimento de uma demo em que o Embree interage com um motor de física para gerar animações fisicamente plausíveis. Espera-se como produto final o código da demo, um tutorial e uma apresentação.

5. **Embree e Bidirectional path tracing** (dificuldade D3)

Pretende-se desenvolver um protótipo básico de um bidirectional path tracer em Embree. Admite-se o suporte de BRDFs extremamente simples (exemplo: difuso e especular ideais).

6. **Melhor convergência usando geradores de números pseudo-aleatórios não correlacionados** (dificuldade D3)

O gerador de números aleatórios distribuído com os tutoriais de Embree não apresenta grande qualidade quando são usadas múltiplas instâncias em várias *threads*. Pretende-se explorar outras propostas presentes na bibliografia conhecida que contribuam para uma melhor convergência e não correlação do trabalho realizado pelas diferentes *threads*.

7. **Modelação de um número de cenas** (pelo menos 5) demonstrando: (dificuldade D1)

- indoor and outdoor;
- número variado de primitivas geométricas e instanciação;
- fenómenos difusos / especulares / *glossy* ;
- número variado de fontes de luz;
- configurações difíceis: *teapot in stadium, light through the key hole*.

8. **Optimização do desempenho recorrendo ao ISPC** (dificuldade D2)

Adaptação do código do *path tracer* usado nas aulas ao ISPC e análise detalhada do respectivo desempenho. O Intel ISPC (<https://ispc.github.io/>) é um compilador de uma variante de C concebido para explorar as unidades de processamento vectorial dos processadores Intel actuais. O Embree foi concebido de raiz com o propósito de interoperar com este compilador.

9. **Optimização do desempenho recorrendo ao *streaming* e/ou aos pacotes de raios** (dificuldade D2)

O Embree suporta processamento de pacotes de raios (1, 4, 8 ou 16 raios) que permitirão maximizar o rendimento das unidades de processamento vectorial se os raios forem coerentes. Simultaneamente, suporta *streams* de pacotes, por forma a manter estas unidades de processamento com altas taxas de ocupação. Pretende-se adaptar o código do *path tracer* desenvolvido nas aulas a estas modalidades de processamento e verificar da respectiva eficiência.

10. **Optimização do desempenho usando múltiplas máquinas** (ex., com MPI no *search*) (dificuldade D3)

A divisão do trabalho por múltiplas máquinas (ou processos) é uma opção óbvia no caso do *ray tracing*. O plano da imagem é dividido em sub regiões (*tiles*) que são processadas pelas diferentes máquinas em paralelo. A comunicação de dados e resultados, bem como a coordenação de todo o paralelismo pode ser conseguido recorrendo a uma biblioteca como o MPI (*Message Passing Interface*). Pretende-se desenvolver uma versão paralela de base para o *path tracer* desenvolvido nas aulas e avaliar o seu desempenho.