



Universidade Federal da Bahia - UFBA

Instituto de Matemática - IM

Departamento de Ciência da Computação - DCC

Curso de Bacharelado em Ciência da Computação

MATA65 - Computação Gráfica

Período: 2018.1

Data: 15/05/2018.

Prof. Antonio L. Apolinário Junior

Estagiário Docente: Rafaela Souza

Roteiro do Laboratório 5 – Visibilidade

Objetivos:

- Reforçar os conceito de **Visibilidade**;
- Analisar a influência das técnicas/algoritmos de visibilidade na formação da imagem de um cenário 3D;
- Desenvolver exemplos práticos, com uso da linguagem *Three.js*, que utilizam os conceitos/técnicas/algoritmos abordados nesse Lab.

Conceitos básicos:

No cálculo de visibilidade da cena, é necessário lidar com a oclusão gerada pelo próprio objeto e entre objetos. A auto-occlusão pode ser resolvida por meio do algoritmo de **backface culling**, a partir da identificação da orientação das faces em relação ao observador. A oclusão entre objetos é tradicionalmente resolvida a partir do algoritmo **z-buffer**, que armazena a menor distancia entre o observador e os polígonos presentes na cena.

Em relação ao controle de visibilidade no *Three.js*, o cálculo de oclusão por *z-buffer* pode ser habilitado a partir da definição do atributo booleano `depthTest`, vinculado ao material do objeto (`THREE.MeshBasicMaterial`, por exemplo). Já o calculo do *backface culling* pode ser habilitado a partir do atributo `side` do material do objeto.

Roteiro:

1. Baixe do Moodle os códigos fonte e as dependências base para esse Laboratório. Descompacte no diretório que será visível pelo servidor *web*.
2. Configure o servidor *web*¹ e execute cada um dos exemplos desse Laboratório.

¹ necessário para o ambiente Windows. Utilize o programa USBWebServer (<http://www.usbwebserver.net/en/>) disponível no repositório da disciplina: <http://homes.dcc.ufba.br/~apolinario/Disciplinas/2017.1/MATA65/USBWebserver%20v8.6.zip>

3. Abra os códigos e analise o exemplo que compõe esse Laboratório. Entenda como a câmera foi parametrizada e como os algoritmos de visibilidade foram implementados.
4. Tomando como base o código desenvolvido no **item 12 do Lab. 4**, substitua o modelo de controle orbital pelo modelo de controle de voo para a câmera perspectiva e navegue pelo cenário. Faça uma navegação pelo cenário com as diferentes opções de *backface culling*. Verifique se houve alguma diferença na visualização do modelo.
5. Utilizando o código desenvolvido no item 4, adicione outro objeto e o posicione dentro da cidade. Ajuste a câmera de forma que este objeto seja visualizado em conjunto com o cenário da cidade. Habilite e desabilite a visualização do *z-buffer* a partir da interface existente. Verifique se houve alguma diferença no modo de visualização gerado com e sem *z-buffer*.