



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: 26/06/2018

Prof. Antonio L. Apolinário Júnior

Estagiário Docente: Rafaela Alcantara

Roteiro do Laboratório 8 – Filtragem Espacial com *ShaderPass*

Objetivos:

- Reforçar os conceitos de filtragem espacial;
- Aprofundar a compreensão do uso de *shaders* e seu funcionamento;
- Entender como o processamento de imagens digitais pode ser feito com a biblioteca *Three.js*.

Conceitos básicos:

A aplicação de uma sequência de filtros em uma imagem é prática comum em Processamento Digital de Imagens. Utilizando *shaders* para esse processamento requer que os resultados de cada etapa do processo de filtragem seja armazenados na forma de uma nova imagem para consumo na etapa seguinte. Esse processo é o que denominamos ***off-screen render***. Ou seja, nesse processo de renderização os fragmentos produzidos pelo *shader* não serão escritos diretamente no *framebuffer*, mas em uma imagem/textura que poderá ser recuperada para novo processamento.

A biblioteca *Three.JS* encapsula todos os detalhes técnicos envolvidos nesse processo em um objeto ***EffectComposer***. Nesse objeto podemos definir uma sequência de *shaders* a serem executados.

Roteiro:

1. Baixe os códigos fonte e as dependências base para esse Laboratório do Moodle. Descompacte no diretório que será visível pelo servidor *web*.
2. Configure o servidor *web*¹ e execute o exemplo desse Laboratório.
3. Abra o exemplo deste laboratório e analise o processo de composição de *shaders* com o *Three.js*.
4. Baseado no exemplo **Compose.html**, crie dois *shaders*: um para conversão da imagem para tons de cinza e outro para aplicação de um filtro de suavização. Aplique-as sequencialmente sobre a imagem original utilizando dois **THREE.ShaderPass**.
5. Com base no código do item 4, troque o filtro de suavização pelo filtro Sobel para detecção de bordas. Analise o resultado da sua aplicação na imagem. Garanta que a visualização dos resultados é correta.

Sobel		
Vertical		Horizontal
-1	0	1
-2	0	2
-1	0	1

6. Modifique o código do item 5 para que o seu shader possa utilizar outros filtros, como o Prewitt.

Prewitt		
Vertical		Horizontal
-1	0	1
-1	0	1
-1	0	1

7. Acrescente um passo a mais no processo de composição de tal modo que o resultado da aplicação dos filtros Sobel ou Prewitt sejam aplicados a imagem original (em tons de cinza) promovendo o *sharpening*. Analise os resultados obtidos.
8. Modifique o código do item 7 para que o processo de *sharpening* seja feito pela técnica de *high-boost*. Permita que o usuário controle o fator de boost.

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/2016.1/MATA65/USBWebserver%20v8.6.zip>