



CENTRO UNIVERSITARIO
DE TECNOLOGÍA Y ARTE DIGITAL

Estudio sobre sistemas de anti-aliasing e implmenetación de anti-aliasing temporal

Hugo Ferrando Seage

U-Tad

septiembre 2018

Tutor: Alberto Sánchez Campos

Resumen

Debido a la resolución limitada de las pantallas al rasterizar gráficos 3D se deforman ciertas líneas y curvas. Este fenómeno se llama aliasing. La demanda de gráficos cada vez más realistas ha propiciado la creación de diferentes técnicas y algoritmos para disimular estos defectos, sin tener que recurrir necesariamente a pantallas de mayor resolución.

El antialiasing se ha vuelto una técnica crucial para mejorar la calidad de imagen en el software con gráficos tridimensionales. Este campo de la computación gráfica lleva años desarrollándose, con nuevas técnicas publicadas continuamente.

Este proyecto tiene como objetivo el análisis de las distintas técnicas existentes en el estado del arte y la implementación de un algoritmo de temporal antialiasing, compatible con motores con deferred shading y comparable en calidad al multisampling, junto a otras propiedades.

Abstract

When rasterizing 3D graphics using computer screens, due to their limited resolution some lines and curves will become deformed. This phenomena is called aliasing. The ever increasing demand for more realistic graphics has driven the creation of new techniques and algorithms to hide these artifacts, without necessarily using higher resolution screens.

Antialiasing has become a crucial technique to advance image quality in graphics software. This field has been explored for years, with new approaches being developed continually.

This projects aims to analyze different antialiasing techniques developed and the implementation of a temporal antialiasing technique, compatible with deferred shading engines and with similar quality to mutlisampling, amongst other benefits.

Índice general

1. Introducción	11
1.1. ¿Qué es el aliasing?	11
1.2. ¿Qué es el antialiasing?	11
2. Planteamiento del problema	13
2.1. Aliasing en bordes de geometría	13
2.2. Aliasing de texturas	13
2.3. Specular Aliasing	13
2.4. Motion Aliasing	13
3. Objetivos	15
4. Estado del Arte	17
4.1. Supersampling Antialiasing	17
4.1.1. Ordered Grid Supersampling	18
4.1.2. Rotated Grid Supersampling	18
4.1.3. Sparse Grid Supersampling	18
4.2. Multisampling	18
4.3. Post Processing Antialiasing	18
4.3.1. Fast Approximate Anti-aliasing	18
4.3.2. Morphological Antialiasing	18
4.3.2.1. Enhanced Subpixel Morphological Antialiasing	18
4.3.3. Temporal Antialiasing	18
5. Desarrollo	19
5.1. Jitter	19
5.2. Motion Vectors	19
5.3. Combinación de texturas	19
5.4. Neighborhood Clamping	19
5.5. Sharpening kernel	19

6. Resultados	21
7. Conclusión	23

Glosario

OpenGL API estándar para el desarrollo de software con gráficos 2D o 3D con aceleración por hardware[4].

rasterizar Conversión de una imagen descrita con alguna primitiva (curvas, vectores, triángulos) a un conjunto de pixels.

Capítulo 1

Introducción

1.1. ¿Qué es el aliasing?

El antialiasing es la deformación de ciertos elementos gráficos al ser rasterizados y plasmados en una pantalla con una resolución finita[1]. Normalmente los bordes de la geometría que tengan ángulos que no se alineen perfectamente con los pixels presentaran bordes de sierra abruptos, que no es fiel a la escena como se vería de forma natural.

Este proyecto se centra en aliasing dentro del campo de la computación gráfica, pero afecta a otras disciplinas, como el procesamiento de señales (por ejemplo al digitalizar señales de audio analógicas). En general, al representar valores continuos en algún medio discreto siempre habrá algún defecto de este tipo[1].

Existen otros tipos de aliasing, como en el interior de texturas o al mover la camera rápidamente.

1.2. ¿Qué es el antialiasing?

El antialiasing es el conjunto de técnicas cuyo objetivo es el de disimular o eliminar estas imperfecciones.

Capítulo 2

Planteamiento del problema

2.1. Aliasing en bordes de geometría

imagen

2.2. Aliasing de texturas

Posibles soluciones: Mip Maps, LoD

2.3. Specular Aliasing

2.4. Motion Aliasing

Explicar motion blur en cámaras

Accumulation motion blur (ps2+) Per pixel motion blur (ps3+) Per object motion blur (ps3+)

Capítulo 3

Objetivos

Este proyecto se ha planteado desde desde dos perspectivas. La primera es la de analizar el estado del arte en el campo de la computación gráfica en el tema de algoritmos de anti-aliasing. El otro objetivo es el de crear una

Capítulo 4

Estado del Arte

4.1. Supersampling Antialiasing

El supersampling consiste en minimizar el aliasing usando imágenes con una resolución mayor a la usada para su visualización (antialiasing espacial). Cada pixel en la pantalla será representado por más de un pixel en la imagen. Para el color final normalmente se hace la media del valor de los colores de los múltiples pixels.

Estas técnicas consumen más ancho de banda y memoria, ya que los buffers tienen que guardar más información por pixel que la imagen original.

La calidad del supersampling viene dado por la cantidad de samples por pixel y la técnica para saber que pixels usar en la imagen con mayor resolución para cada pixel de la pantalla.

Figura 4.1: Ordered Grid Super-Sampling[2]

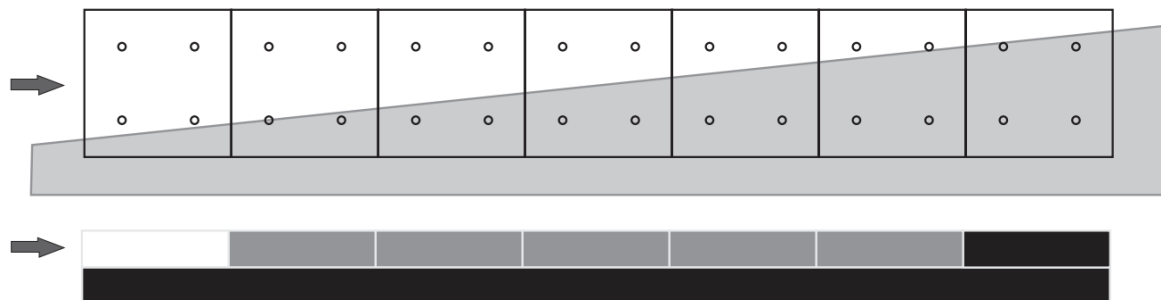
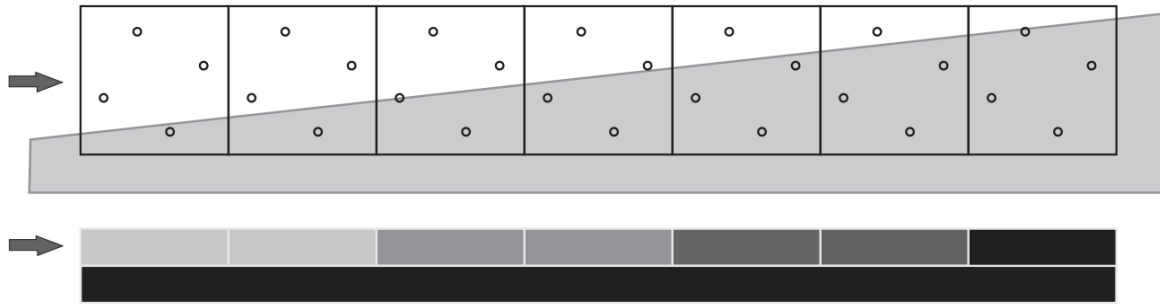


Figura 4.2: Rotated Grid Super-Sampling[2]



4.1.1. Ordered Grid Supersampling

4.1.2. Rotated Grid Supersampling

4.1.3. Sparse Grid Supersampling

4.2. Multisampling

4.3. Post Processing Antialiasing

Este tipo de antialiasing ha ganado popularidad durante los últimos años, debido en parte al uso del deferred shading en los últimos motores gráficos.

4.3.1. Fast Approximate Anti-aliasing

El FXAA[3] es un sistema de anti-aliasing de post procesado desarrollado por NVIDIA.

4.3.2. Morphological Antialiasing

mlaa

4.3.2.1. Enhanced Subpixel Morphological Antialiasing

smaa

4.3.3. Temporal Antialiasing

El temporal antialiasing es un tipo de antialiasing de post procesado, pero que amortiza el coste computacional usando múltiples samples en multiples frames. Esto presenta unas complicaciones a la hora de escoger samples en frames anteriores en imágenes no estáticas.

Capítulo 5

Desarrollo

Para este proyecto se ha integrado un sistema de anti-aliasing temporal en un motor de gráficos 3D que usa OpenGL con API gráfica.

5.1. Jitter

5.2. Motion Vectors

5.3. Combinación de texturas

5.4. Neighborhood Clamping

5.5. Sharpening kernel

Capítulo 6

Resultados

Figura 6.1: Anti-aliasing off



Figura 6.2: Temporal anti-aliasing con 8 samples

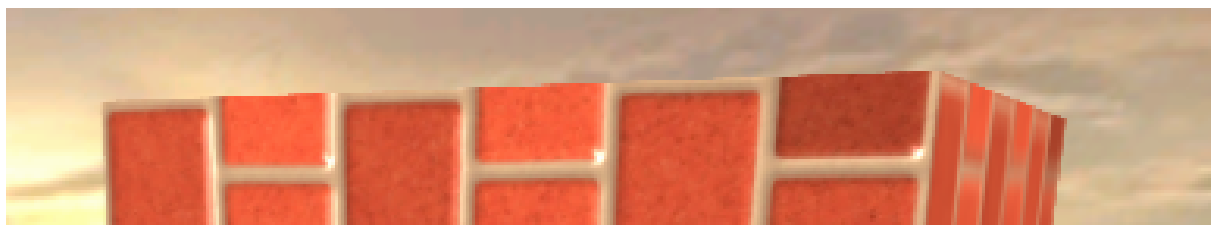
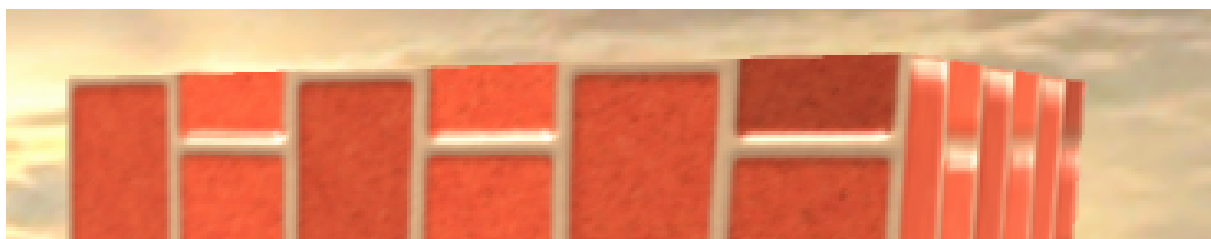


Figura 6.3: Temporal anti-aliasing con 16 samples



Capítulo 7

Conclusión

Bibliografía

- [1] D. P. Mitchell y A. N. Netravali, «Reconstruction Filters in Computer-graphics», *SIGGRAPH Comput. Graph.*, vol. 22, n.º 4, págs. 221-228, jun. de 1988, ISSN: 0097-8930. DOI: 10.1145/378456.378514. dirección: <http://doi.acm.org/10.1145/378456.378514>.
- [2] K. Beets y D. Barron, «Super-sampling Anti-aliasing Analyzed», 2000.
- [3] T. Lottes, «FXAA. NVIDIA White Paper», inf. téc., 2009.
- [4] J. Kessenich, G. Sellers y D. Shreiner, *OpenGL®Programming Guide: The Official Guide to Learning OpenGL®, Version 4.5 with SPIR-V*, 9.ª ed. Addison-Wesley Professional, 2016, ISBN: 9780134495491.

Índice de figuras

4.1. Ordered Grid Super-Sampling[2]	17
4.2. Rotated Grid Super-Sampling[2]	18
6.1. Anti-aliasing off	21
6.2. Temporal anti-aliasing con 8 samples	22
6.3. Temporal anti-aliasing con 16 samples	22

Índice de cuadros

