

# An Introduction to Foundation of Computer Graphic

Yunshui Jin



- **OpenGL** (Open Graphics Library) is a standard specification defining a cross-language, multi-platform **API** for writing applications and **simulating physics**, that produce 2D and 3D computer graphics.

# History

**1980s**

**Early 1990s** Silicon Graphics (SGI) IRIS GL API

OpenGL standard, giving software developers a higher level platform for 3D-software development.

**1992**, OpenGL 1.0, OpenGL architectural review board (OpenGL ARB), the group of companies that would maintain and expand the OpenGL specification for years to come. OpenGL overcame this problem by providing support in software for features unsupported by hardware, allowing applications to use advanced graphics on relatively low-powered systems.

**1994**, OpenGL++

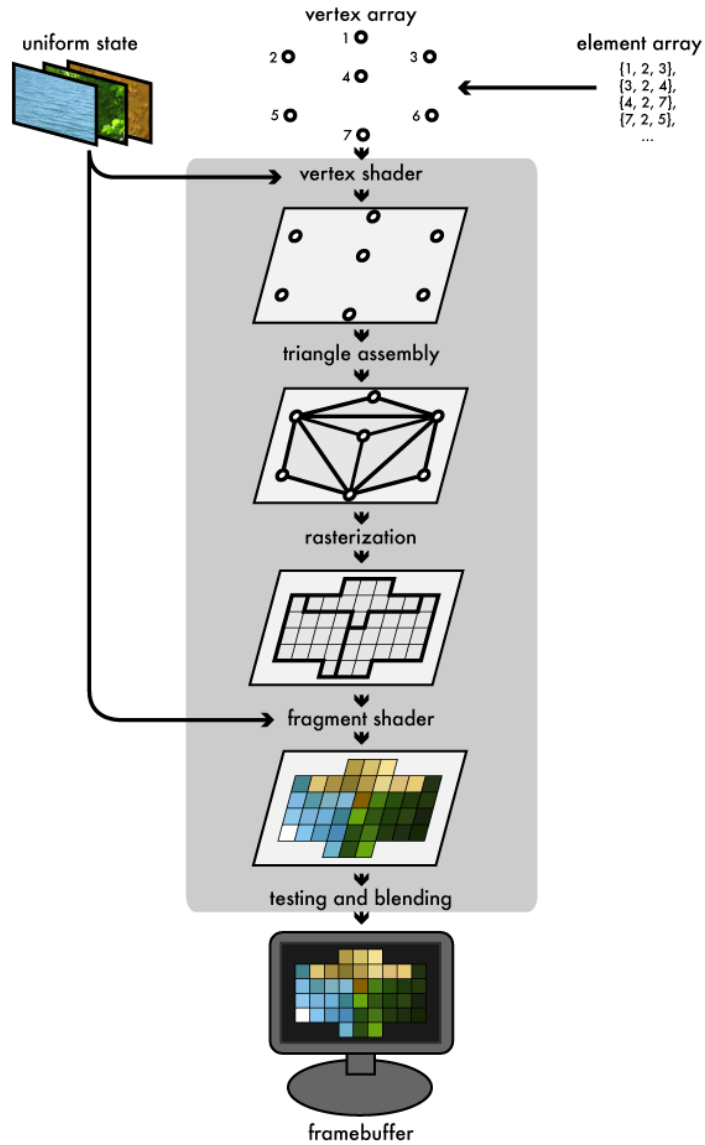
**1995**, Direct3D, Microsoft and SGI initiated the Fahrenheit project, which was a joint effort with the goal of unifying the OpenGL and Direct3D interfaces. But failed.

**2006**, OpenGL 2.1

**2011**, OpenGL 4.2

*OpenGL extension viewer*

# The Graphics Pipeline



Camera

Ant aliasing

Z-buffer

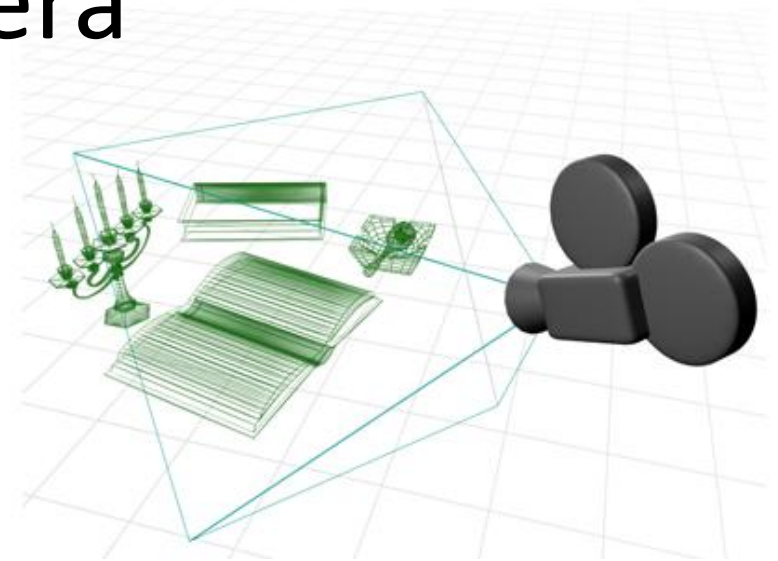
Vertex color

Texture

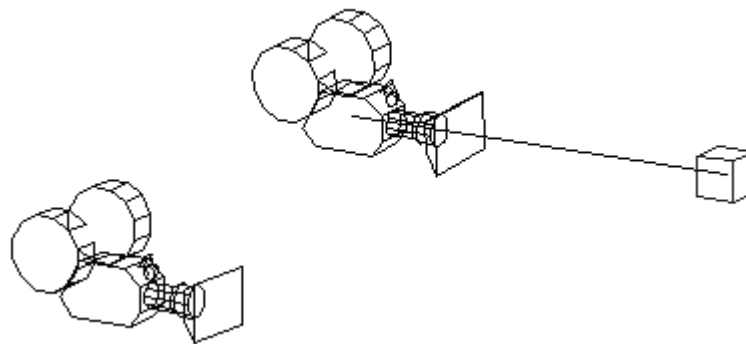
Shader

# camera

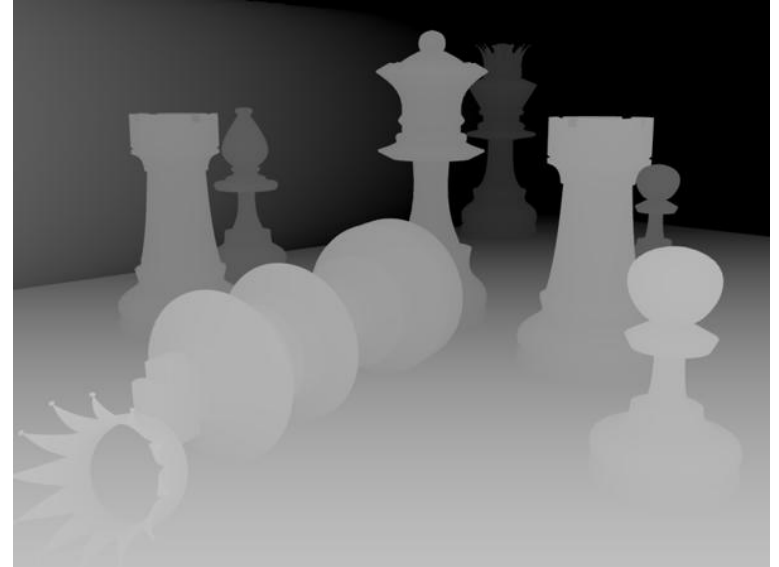
- Basic camera
- Object Inspection camera
- Walking through camera



- Basic matrix(position rotation size)
- Look at matrix
- Parent matrix



# Z-buffer



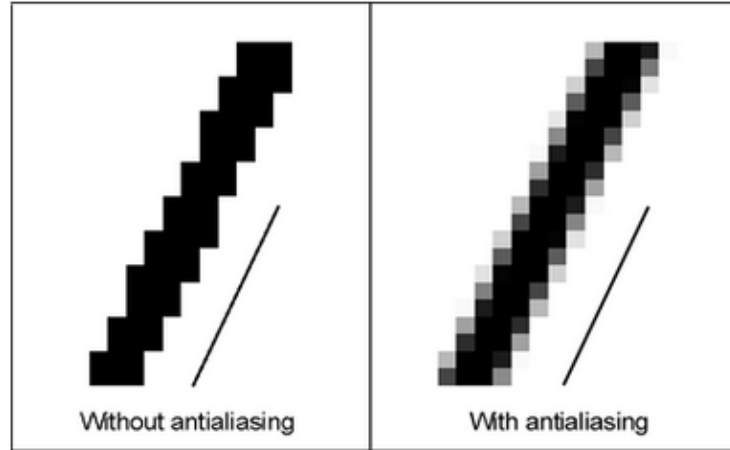
- Real-time DOF
- Clipping plane

# Vertex color



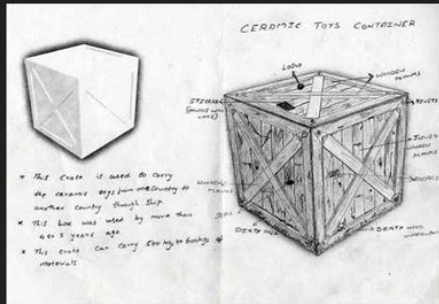
- Mainly for lighting purpose
- Vertex color Vs Light map

# Anti aliasing



- Geometry anti-aliasing
  - Many algorithm

# Texture



Concept Sketch

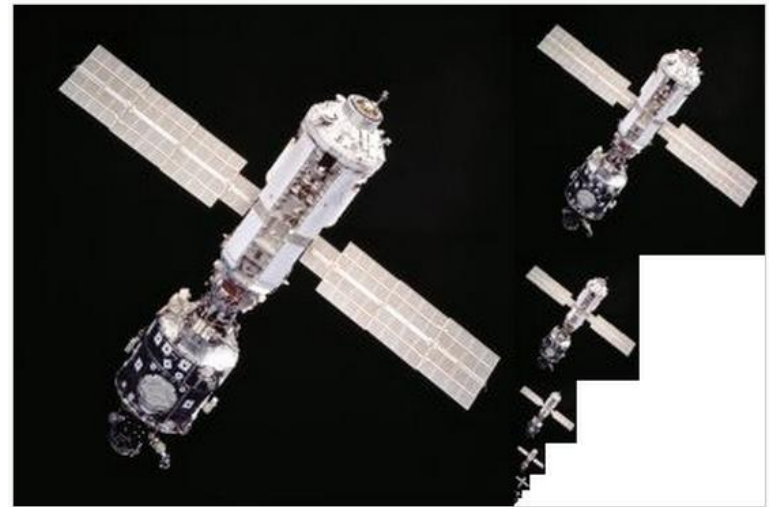
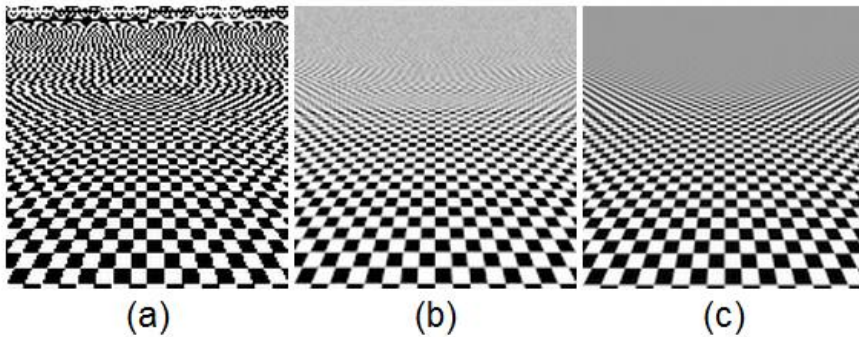


Texture



Screenshot from 3ds Max

- Mapping UVW to the modeling



- Texture anti-aliasing(mipMap)
- DDS Format <http://developer.nvidia.com/nvidia-texture-tools-adobe-photoshop>

# Shader

- GLSL(OpenGL Shading Language) & HLSL( high-level shading language) to achieving amazing shader
- Nvidia Fx composer & AMD Render Monkey

