KAIST

OPENGL tutorial 2

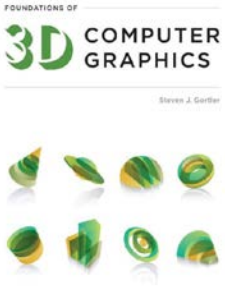
Joo Ho Lee

VISUAL COMPUTING Lab

Announcement

KAIST

- Please buy this book.



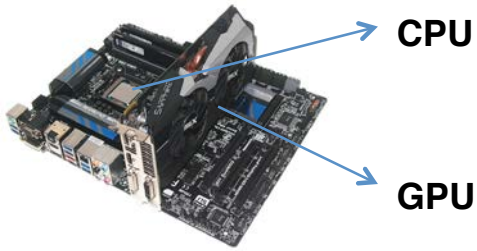
Steven J. Gortler, MIT PRESS, 2012

VISUAL COMPUTING Lab

Computer Structure

KAIST

- A computer has two processing units
  - Central processing unit: CPU
  - Graphics processing unit: GPU

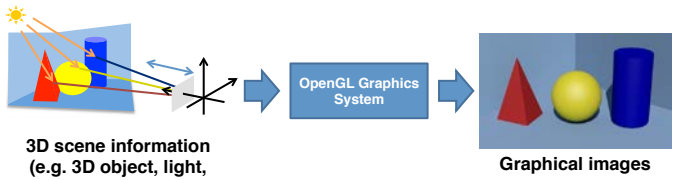


VISUAL COMPUTING Lab

OpenGL graphics system

KAIST

- The system working on **GPU** to **produce high-quality graphical images, specifically color images of three-dimensional objects.**

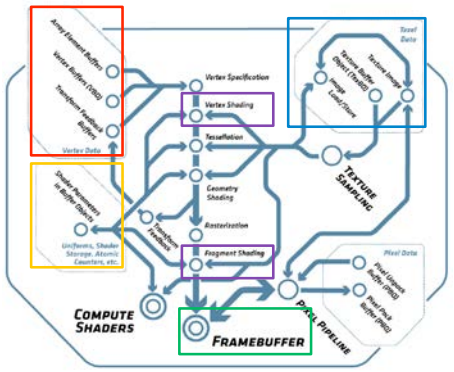


3D scene information (e.g. 3D object, light, color, camera)

Graphical images

VISUAL COMPUTING Lab

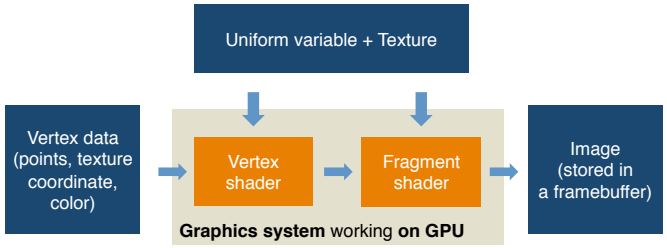
OpenGL graphics system overview



Vertex data
Shader
Framebuffer
Texture
Uniform variable

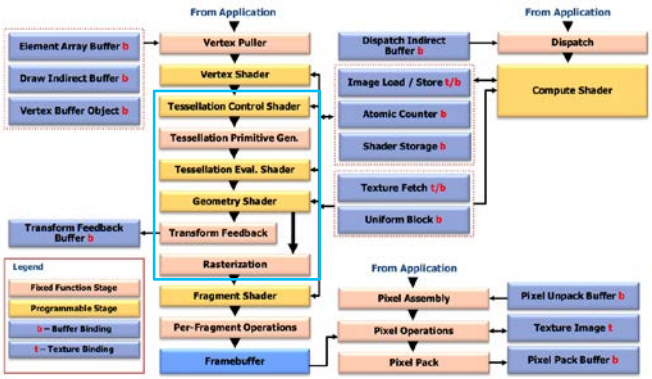
VISUAL COMPUTING Lab

Brief graphics system for programming



VISUAL COMPUTING Lab

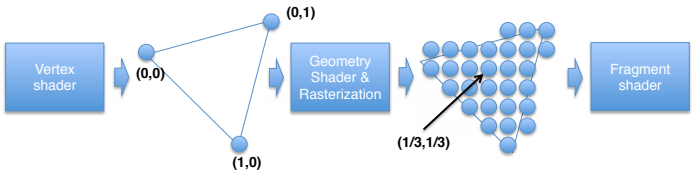
Detail of openGL graphics system



VISUAL COMPUTING Lab

Dataflow in the OpenGL pipeline

- Vertex shader
  - Set and process data for each vertex
- Rasterization
  - Interpolate values of each pixel from vertices
- Fragment shader
  - Pixel-wise computation



8

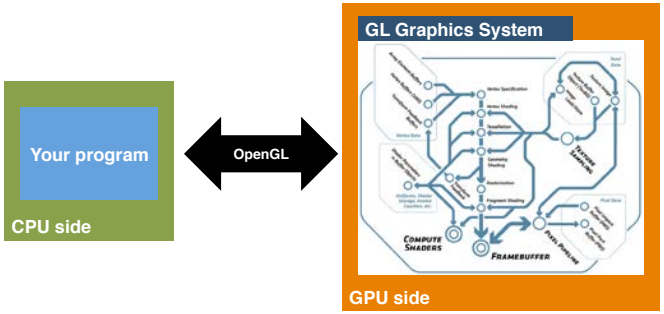
VISUAL COMPUTING Lab

Why we need OpenGL?

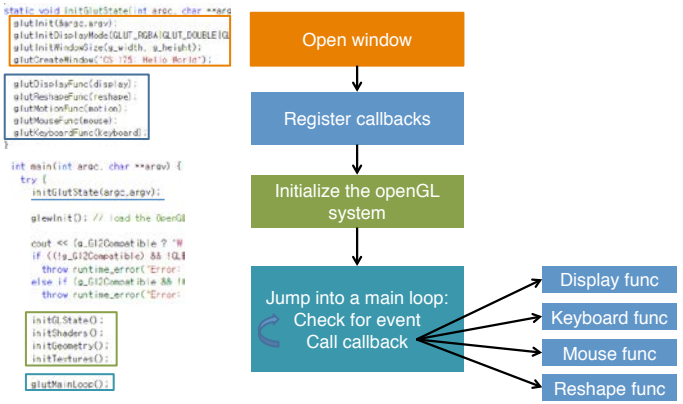
- How we pass 3D scene data to the OpenGL graphics system?
- How we change the mode (state) of the graphics system?
- How we launch the graphics system to render the image?

Overview of your openGL program

- Using openGL, your program control and interact with the OpenGL graphics system.

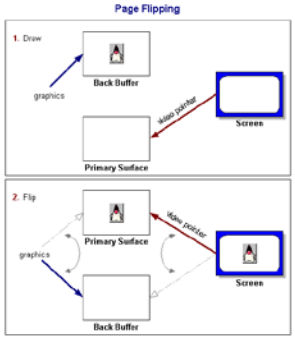


openGL Program workflow



Double buffering

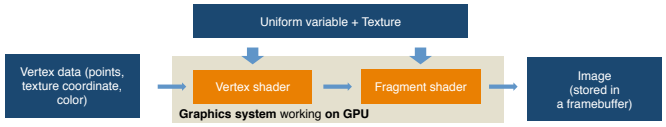
- glutInitDisplayMode(GLUT_DOUBLE)
- Page Flipping: glutSwapBuffer()



Passing data to the openGL graphics system – Uniform variable

KAIST

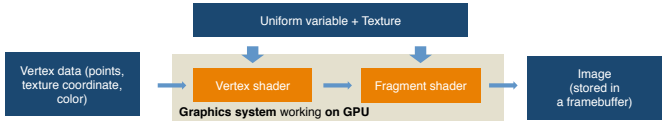
- First, get a location of a uniform variable
  - glGetUniformLocation
- Second, specify the value of a uniform variable
  - glUniformXX



VISUAL COMPUTING Lab

Passing data to the openGL graphics system – vertex data

KAIST

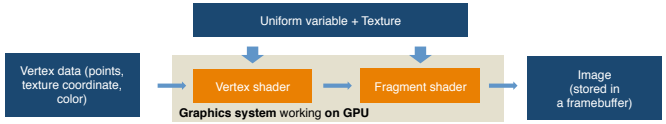


VISUAL COMPUTING Lab

Where is a start button for rendering?

KAIST

openGL graphics system starts to operate!

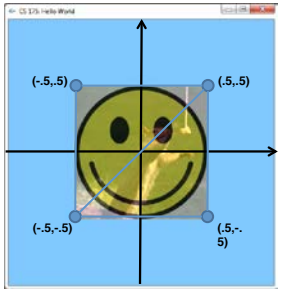


VISUAL COMPUTING Lab

Homework 1

KAIST

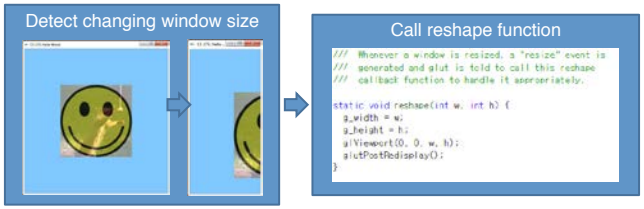
- Geometric overview



VISUAL COMPUTING Lab

Reshaping

KAIST



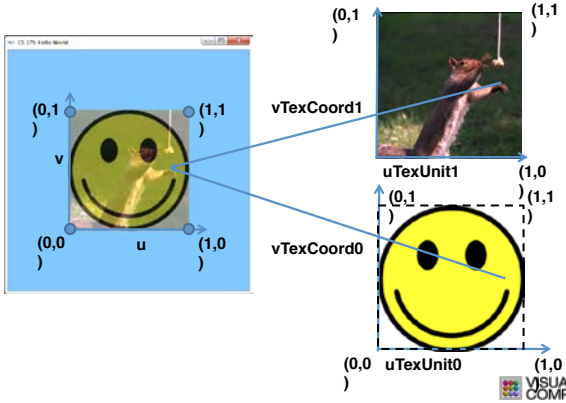
- glViewport sets the width and height of the viewing box.
- GlutPostRedisplay() calls the display callback function.

VISUAL COMPUTING Lab

Texture mapping

KAIST

- texture(uTexUnitX, vTexCoordX)

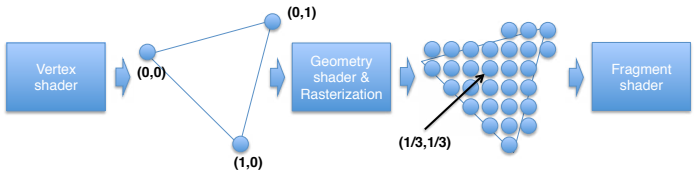


VISUAL COMPUTING Lab

Texture mapping

KAIST

- In a vertex shader
  - Set a texture coordinate for each vertex
- In a fragment shader
  - As an input of the fragment shader, we get an interpolated texture coordinate for each pixel.



19

VISUAL COMPUTING Lab

Tip

KAIST

- The openGL functions are well specified in OpenGL documentation.
- Just google it or read a text book.

20

VISUAL COMPUTING Lab