ICG Syllabus

1. Introduction
   1. What is interactive computer graphics, GUI?
   2. What is virtual reality, and augmented reality?
   3. What is OpenGL/WebGL and shading language?

2. Viewing in 3D and Graphics Programming
   1. Parallel projection, Perspective projection etc.
   2. First program: Sample codes in WebGL

3. Basic raster graphics algorithms for drawing 3D primitives
   1. 3D primitives (Points and Vectors)
   2. Geometrics Transformations, Viewing in 3D, 3D clipping

4. Visible-surface determination
   1. Z-buffer algorithm
   2. Binary space partition trees (BSP)
   3. Visible-line determination

5. Illumination and Shading
   1. The Phong Reflection Model
   2. Flat shading, Gouraud shading, Phong shading
   3. Bilinear interpolation for color and vertex normal interpolation
Syllabus

6. Implementation of a Renderer
   Graphics Pipeline: from modeling to display
   Implementation by WebGL, Unity3D (game engine)

7. Global rendering--Photo-realistic graphics
   1. Recursive ray tracing
   2. Volume rendering: marching cubes method etc.
   3. Radiosity method: progressive refinement approach

8. Curves and surfaces: Bezier curves, B-Splines, Spline surfaces

9. Graphics Hardware and Graphics Processing Unit (GPU), GPGPU, APU, Deep Learning Acceleration

10. Implementation and practice
    1. Graphics data set
    2. How to quickly construct the 3D model data of the world for 3D graphics/AR/VR?
    3. Modeling methods and tools

11. Demonstration and Term Projects
Learning experience

• **Engage Students Immediately with 3D Material:** A top-down, programming-oriented approach allows for coverage of engaging 3D material early in the course so students immediately begin to create their own graphics.

• **Introduce Computer Graphics Programming with WebGL and JavaScript:** WebGL is not only fully shader-based—each application must provide at least a vertex shader and a fragment shader—but also a version that works within the latest web browsers.

• **Create games by example:** to modify an existing program in Unity3D, and then create an interesting game/animation of your own.