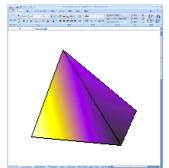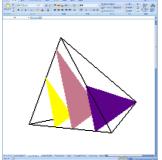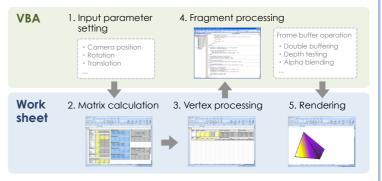# An Educational System of Scientific Visualization Techniques Using Microsoft Excel Spreadsheets

Naohisa Sakamoto, Koji Koyamada
Kyoto University, Japan

Rendering results (left: boundary faces, right: isosurfaces)

VBA
1. Input parameter setting
   · Camera position
   · Rotation
   · Translation
   ...
4. Fragment processing
   Frame buffer operation
   · Double buffering
   · Depth testing
   · Alpha blending
   ...

Work sheet
2. Matrix calculation
3. Vertex processing
5. Rendering

System architecture

Main characteristics of our system

1. Users can intuitively understand the table-driven algorithm that uses look-up tables, such as in isosurface extraction, by using the worksheet.
2. Because the rendering processes can be visually shown on the worksheet in our system, users can learn the rendering pipeline, which is implemented as a fixed processing stage sequence on a GPU, and is hard to learn using only graphics APIs such as OpenGL.
3. By using our Excel-based system, users can efficiently learn that basis of computer graphics and scientific visualization techniques by using commodity PCs.

Frame buffer
A frame buffer can be created by treating a cell as a pixel.

Cell(5,2).Interior.color = RGB(255,0,0)

Fragment processing
By filling in the cells, which can be regrads as a fragment in OpenGL, on the frame buffer worksheet, it is possible to draw a polygon on the worksheet.

Vertex processing
Transformation matrices for the vertex transformation, such as the modelview matrix and projection matrix in OpenGL, can be calculated on the worksheet by using only standard worksheet functions.
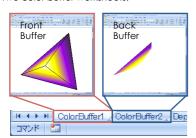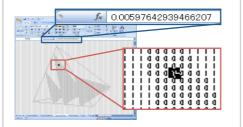
Vertex transformation

Double buffering
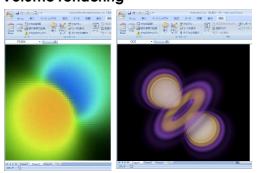A double buffering process is performed for the animation rendering by switching two color buffer worksheets.

Front Buffer

Back Buffer

ColorBuffer1 ColorBuffer2 Dep

Depth testing
Hidden polygon removal can be performed with a Z buffer algorithm implemented in VBA by using the depth buffer worksheet.

0.00597642939466207

Volume rendering