Overview of Graphics Systems

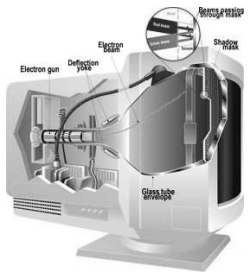
Overview of Graphics Systems

- Graphics Hardware
  - Output / Display device
  - Input device
- Graphics Software
  - OpenGL
  - FLTK /GLUT

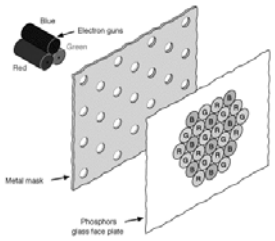
Output/Display Device

- Color Cathode-ray-tube (CRT) monitors
- Liquid Crystal Displays (LCD)
- Plasma Display Flat Panels.
- Virtual-Reality Systems.
- Color Printers.

Color CRT Monitors



Color CRT Monitors



Liquid Crystal Displays (LCD)

Plasma Display Panels

Virtual Reality Systems

Virtual Reality

= Real time graphics

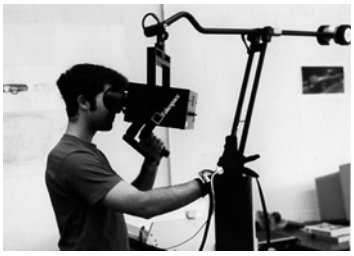
+ Immersion

+ Interaction

+ other perceptive feedback (touch, sound, smell...)

Virtual Reality Systems

Virtual Reality Systems



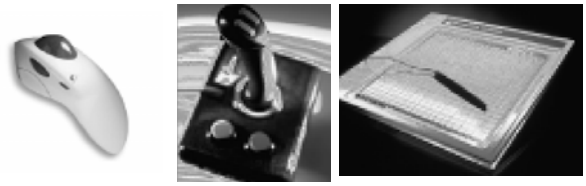
Virtual Reality Systems

Input Devices

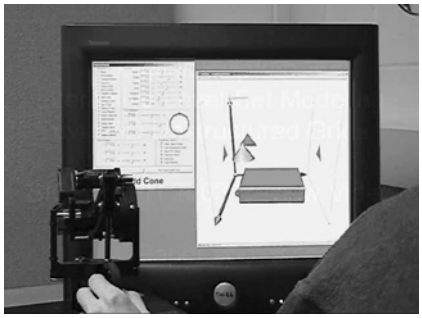
- Keyboards
- Mouse
- Trackball
- Joystick
- Touch pad
- Image Scanners
- 3D laser range scanner.

Keyboard & Mouse

Trackball, Joystick, Touch Pad

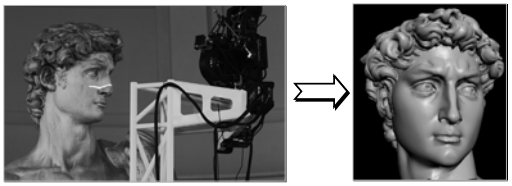


Haptics Device (PhanTom 1.0)



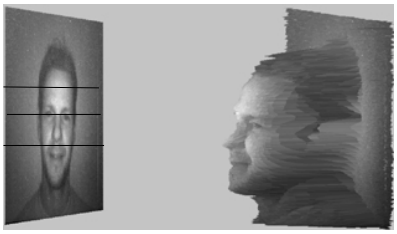
3D Laser Range Scanner

3D Laser Range Scanner



3D Camera

Digital Fringe Projector



OpenGL

- Most widely used 3D graphics Application Program Interface (API).
- Truly open, independent of system platforms.
- Reliable, easy to use and well-documented.
- Default language is C/C++.

OpenGL

- The **`GL`** library is the core OpenGL system:
  - modeling, viewing, lighting, clipping
- The **`GLU`** library (GL Utility) simplifies common tasks:
  - creation of common objects (e.g. spheres, quadrics)
  - specification of standard views (e.g. perspective, orthographic)
- The **`GLUT`** library (GL Utility Toolkit) provides the interface with the window system.
  - window management, menus, mouse interaction

OpenGL

- To create a red polygon with 4 vertices:

```
glColor3f(1.0, 0.0, 0.0);
glBegin(GL_POLYGON);
    glVertex3f(0.0, 0.0, 3.0);
    glVertex3f(1.0, 0.0, 3.0);
    glVertex3f(1.0, 1.0, 3.0);
    glVertex3f(0.0, 1.0, 3.0);
glEnd();
```

- **`glBegin`** defines a geometric primitive:

```
GL_POINTS, GL_LINES, GL_LINE_LOOP, GL_TRIANGLES,
GL_QUADS, GL_POLYGON…
```

- All vertices are 3D and defined using **`glVertex`**

FLTK

- Fast Light Tool Kit (FLTK)
- www.fltk.org
- C++ oriented
  - A set of UI classes such as Window, box, etc.
- Can mix use with GLUT
- FLUID: fast light UI Designer
  - Fast creation of GUI
  - Automatically writes parts of GUI code from a graphical spec
  - Good for elaborate interfaces

Comments on Programming

- OpenGL plus Glut
  - Simple, easy to program, limitations
- OpenGL plus FLTK
  - Cross platform, more powerful
- OpenGL plus Visual C++
  - Super!
  - Only run under windows system

Case Study
(chalk talk)