CSE-421

Computer Graphics and Animation

Lecture 01

Introduction

Md. Shahid Uz Zaman

szaman22.ruet@gmail.com

Syllabus

4th YEAR EVEN SEMESTER

CSE 4000 — Contact Hours/Week: 6
Project / Thesis II — Credits:3.00
Prerequisite: None
Continuation of project/thesis topic under taken in CSE 4000.

CSE 4201 — Contact Hours/Week: 3
Computer Graphics and Animations — Credits:3.00
Prerequisite: None

Introduction: History, Application of Computer Graphics (Computer Aided Design Animation), A Survey of Graphics I/O Devices and Types.

Graphics Software Design: Survey of Desired Function, Toward a Universal Graphic Language. Display Files, Databases for Pictorial Applications.

Graphics Techniques: Point-Plotting Techniques, Line Drawing, Geometric Transformations, Windowing and Clipping, Raster Graphics.

Hardware for Computer Graphics: Typical Small and Large System, Graphic Terminals, Plotters, Graphic Display Processors, Device Independent Graphics Systems.

Graphics Software: A Simple Graphic Package, Segmented Display Files, Geometric Models, Picture Structure.

Interactive Graphics: Input Techniques, Event Handling, Three-Dimensional Graphics, Curves and Surfaces, 3-D Transformation.

Hidden Surface Problem: Back Face Removal, Hidden-Line Removal Curved Surfaces, Describing Points, Lines and Polygons, Some Hints for Building Polygonal Models, Color Perception, RGBA and Color Index Mode, Dithering, Blending, 3-D Blending with the Depth Buffer, Antialiasing, Fog, Fog Equations, OpenGL ARB.

API Specifies: Data Types, Function Naming Conventions, Platform Independence, Drawing Shapes with OpenGL, Animation with OpenGL and GLUT.

Drawing in Space: Lines, Points and Polygons.
Co-ordinate Transformations: Understanding Transformations, Matrix Munching Projections, Matrix Manipulation Color Lighting and Materials, Texture Mapping.

Syllabus

Hidden Surfaces: Depth comparison, Z-buffer algorithm, Back face detection, BSP tree method, the Printer's algorithm, scan-line algorithm; Hidden line elimination, wire frame methods, fractal - geometry.

Color and Shading Models: Light & color model; Interpolative shading model; Texture.

Books Recommended:

1.	Donald Hearn and M. Pauline Baker	:	**Computer Graphics**, *Prentice Hall*
2.	Steven Harrington	:	**Computer Graphics: A Programming Approach**, *McGraw-Hill College*
3.	F. S. Hill	:	**Fundamentals of Computer Graphics**, *Prentice Hall*
4.	Plastock and Kalley	:	**Computer Graphics**, *Mcgraw-hill*
5.	Zhigang Xiang & Roy Plastock	:	**Computer Graphics**, *Mcgraw-hill*

Computer Graphics

Computer Graphics

Computer

Raw Data

process

Meaningful Data

Graph

Geometrical

Object

Point

Line

Circle

Ellipse

Pics

In 1963, Ivan Sutherland developed **Sketchpad** for line drawing by a computer

Definition

Computers...

accept, process, transform and present information.

Computer Graphics...

involves display, manipulation and storage of pictures and experimental data for proper visualization using a computer.

Typical Graphics System

Computer Graphics includes:

- Representation
  - How a geometric figure is to be displayed
- Manipulation
  - Applying algorithm
- Storage

- **Requirements**

  – Typical Graphics System comprises of a host computer with support of fast processor, large memory, frame buffer and

  - Display devices (color monitor)
  - Input devices (mouse, keyboard, joystick, touch screen, trackball)
  - Output devices (LCD panels, laser printers, color printers, Plotters etc.)
  - Interfacing devices (Video I/O, TV interface, game controller

Typical Application Areas

- **GUI**
  - Menus, cursors, icons, dialog box, scroll bar buttons, valuators, grids, sketching, 3D drawing
- **Plotting in Business**
  - Pi-chart, 2D or 3D graphs, growth rate, expenditure profit curves
- **Office Automation**
  - MS Word, MS Excel, MS PowerPoint, Scanning
- **Desktop publishing**
  - Designing documents

Typical Application Areas

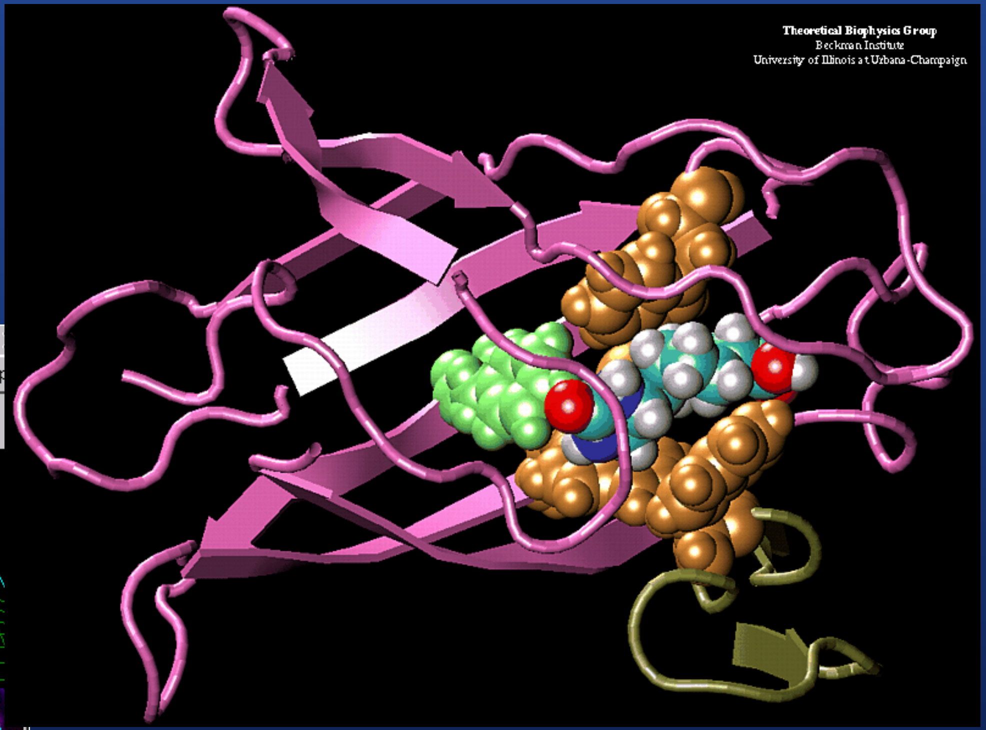
- **Plotting Science and Technology**
  - 2D or 3D graph, line drawing, surface plot, contour plots, pi chart etc.
- **Web/business/commercial publishing/advertisement**
  - On the web, commercial shoot outs, banner
- **CAD/CAM design**
  - VLSI, circuit design, architectural and civil engg. drawings
- **Scientific Visualization**
  - Nuclear explosion, biological or mechanical sciences

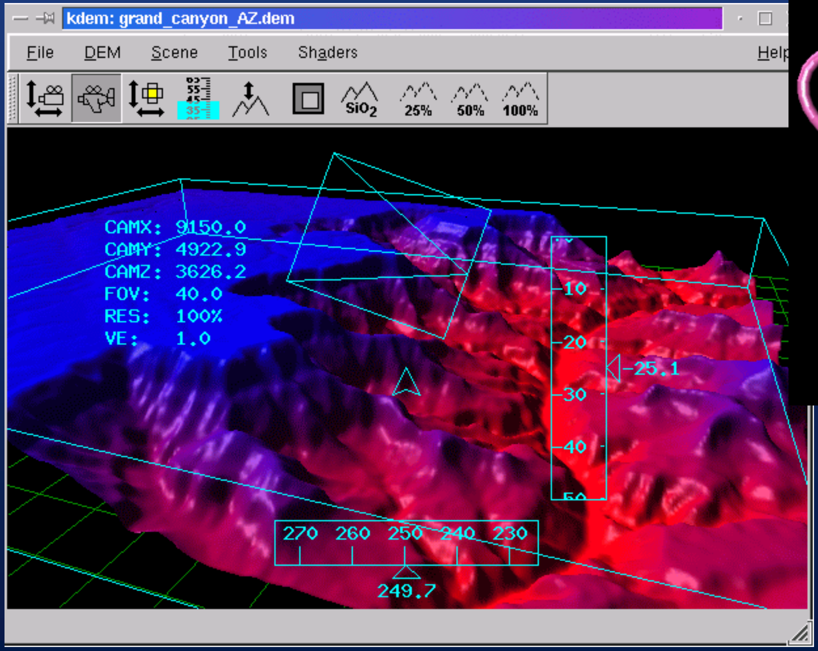
Typical Application Areas

- **Entertainment**
  - Movies, TV advertisements, Games etc.
- **Simulation studies**
  - Space shuttle, auto pilots, sports
- **Cartography**
  - Weather maps, Population density maps, Contour maps
  - Geographical maps Oceanographic maps
- **Multimedia**
  - Text, audio, video graphics
- **Virtual Reality**
  - Games, animation

Application Examples

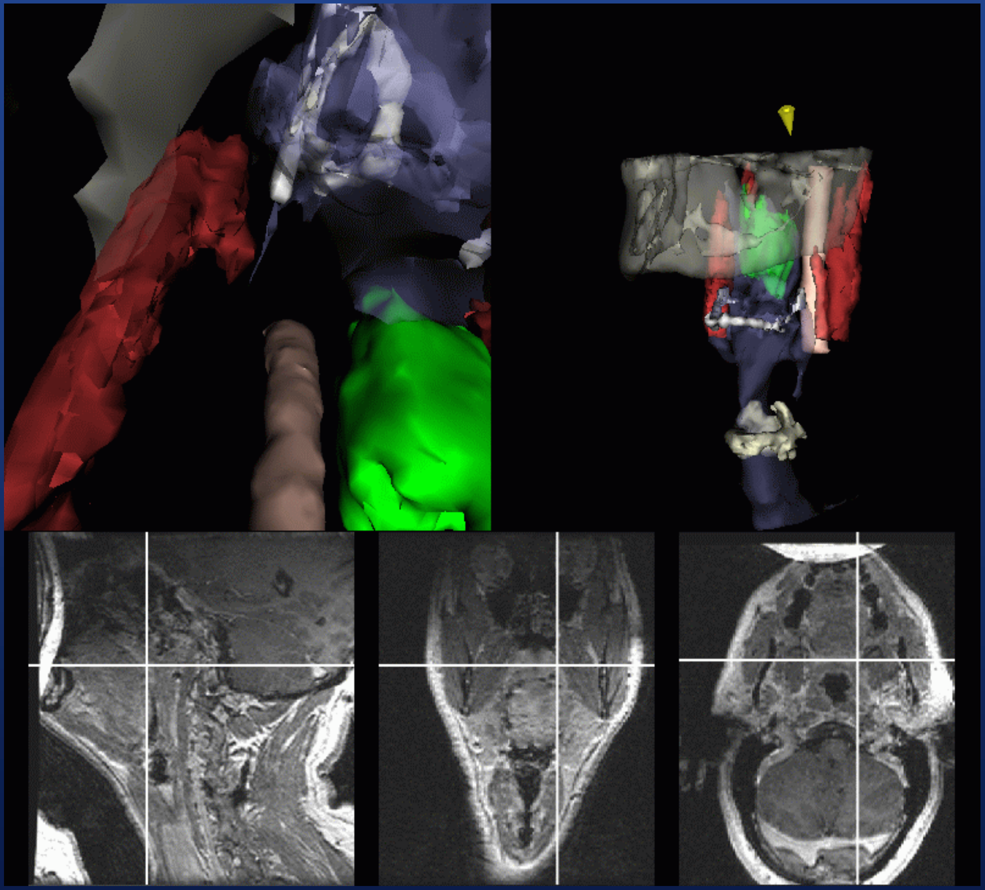
Scientific Visualization and Analysis

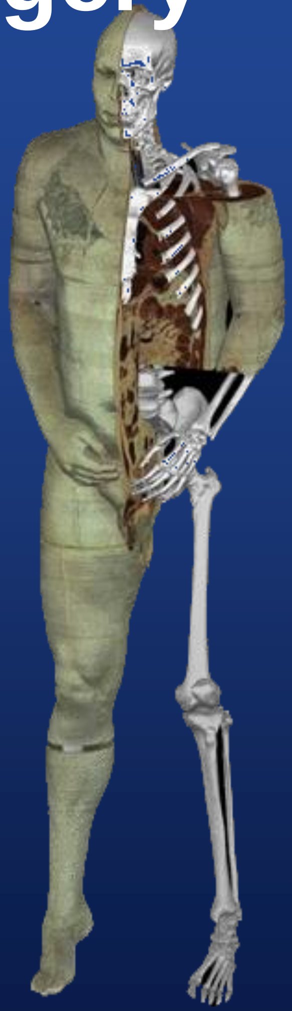




ART AND DESIGN

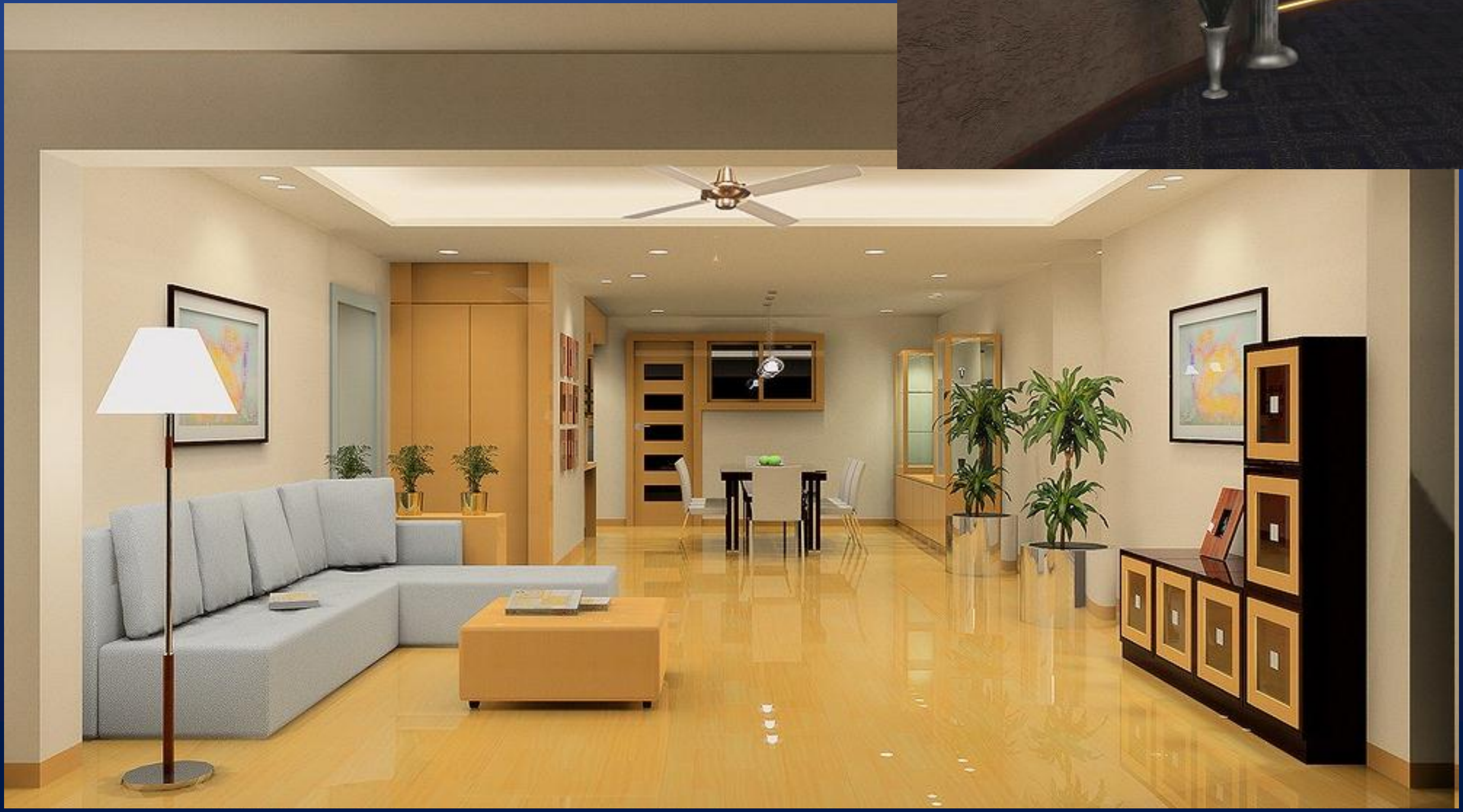
Medicine and Virtual Surgery





The Visible Human project

Room Layout Design and Architectural Simulations

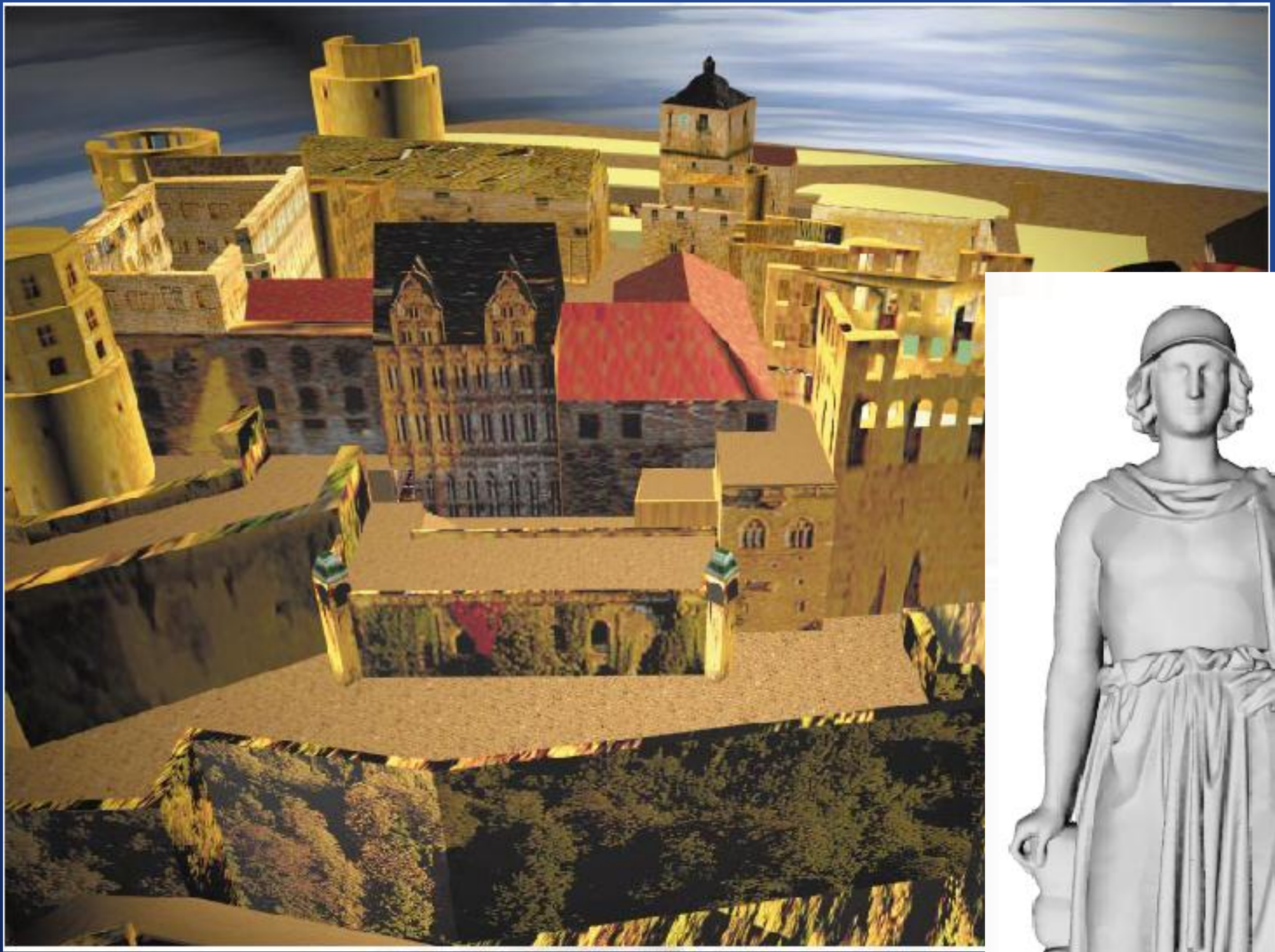


Layout Design & Architectural Simulations

Layout Design & Architectural Simulations

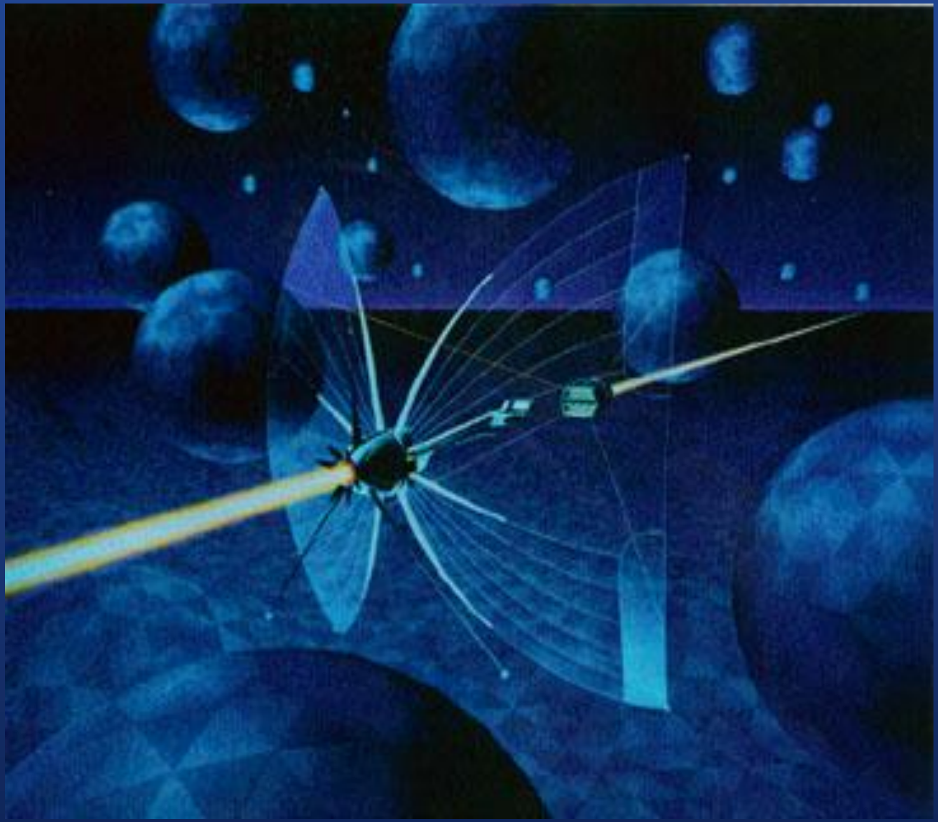
Layout Design & Architectural Simulations

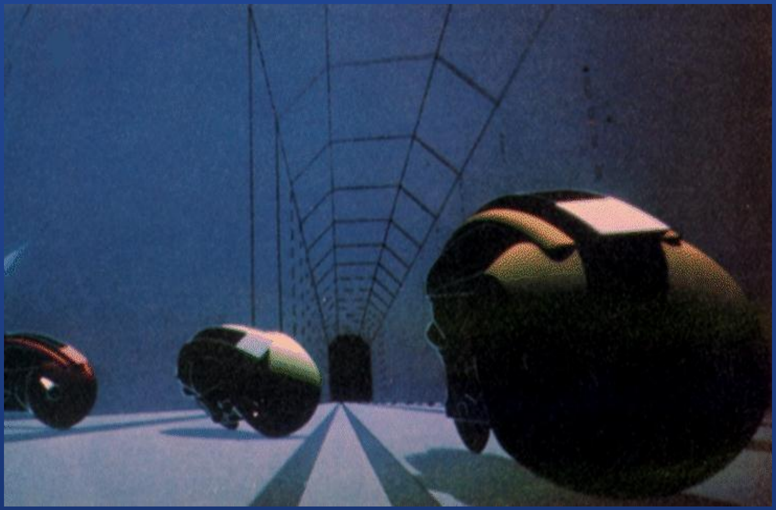
Layout Design & Architectural Simulations

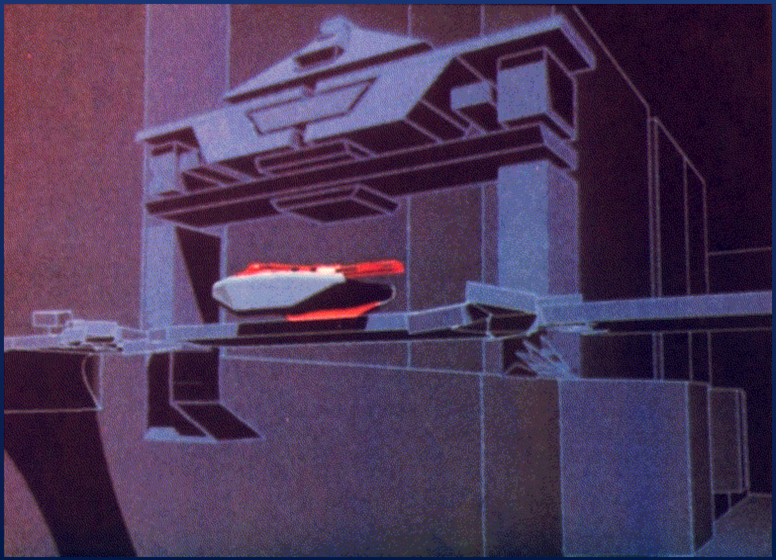


History and
cultural heritage

Movies





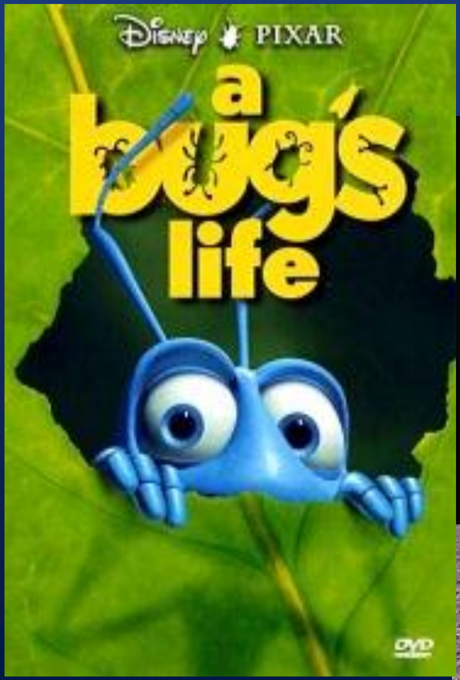


Tron (1980)

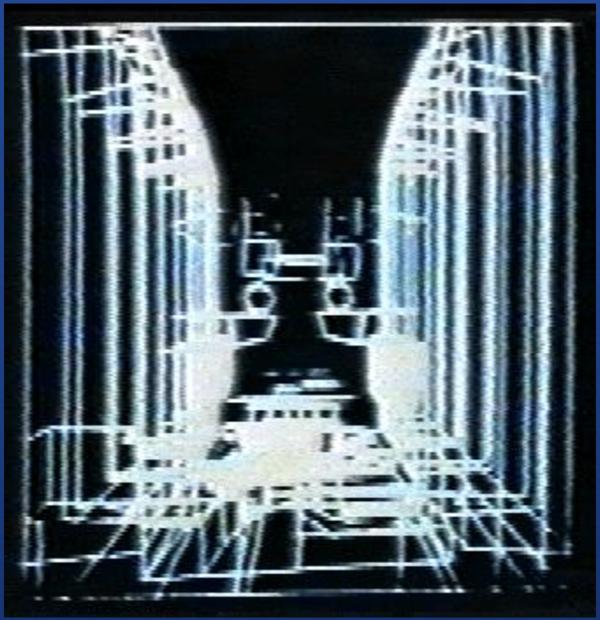
First time computer graphics were used for live action sequences.

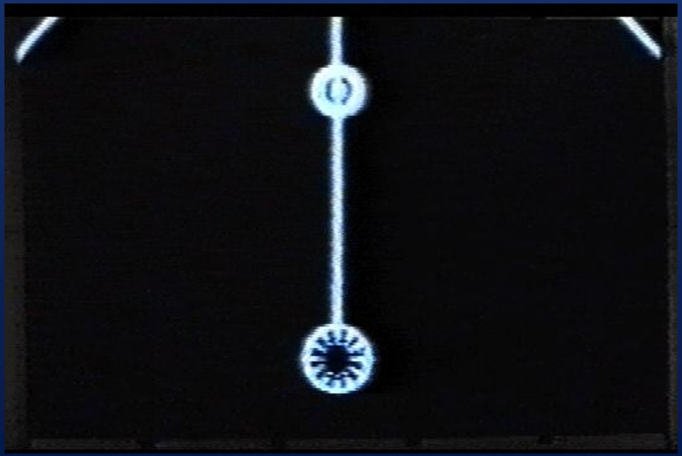
Movies

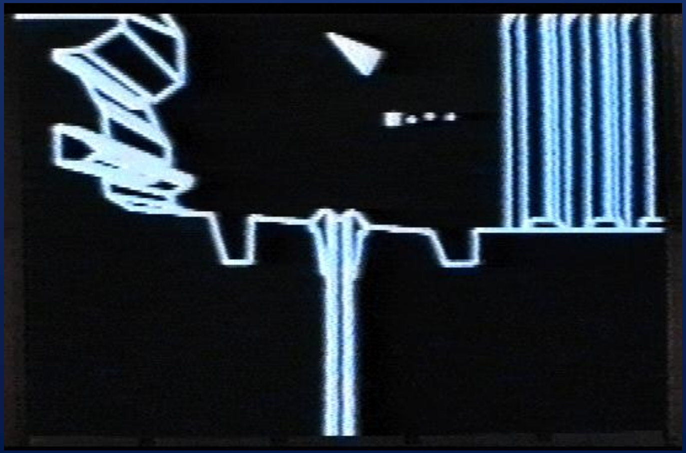
Fully computer generated animated features



Movies

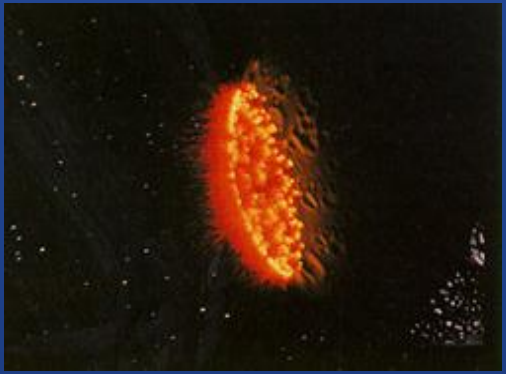




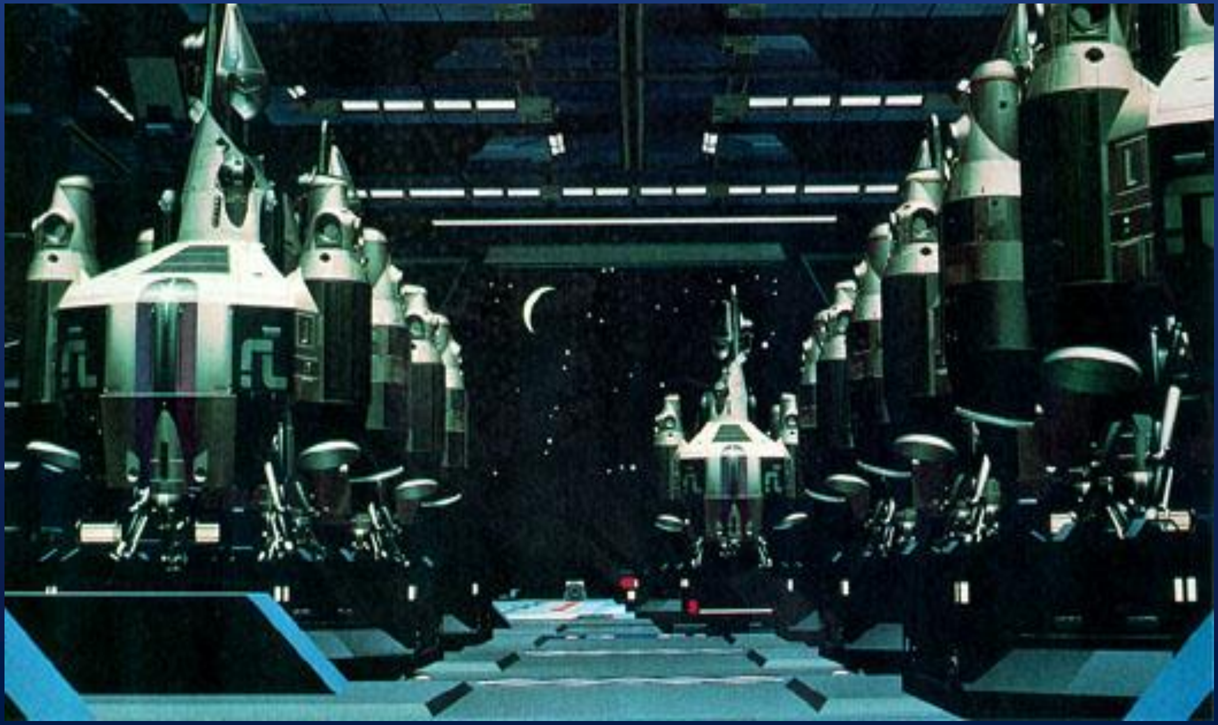


Star Wars (1977)

Movies


Star Trek II: The Wrath of Khan, genesis effect (1983)



The Last Starfighter (15 minutes) (1982)

Movies

Special Effects… in Live Action Cinema

Jurassic park, Titanic and T2

Movies

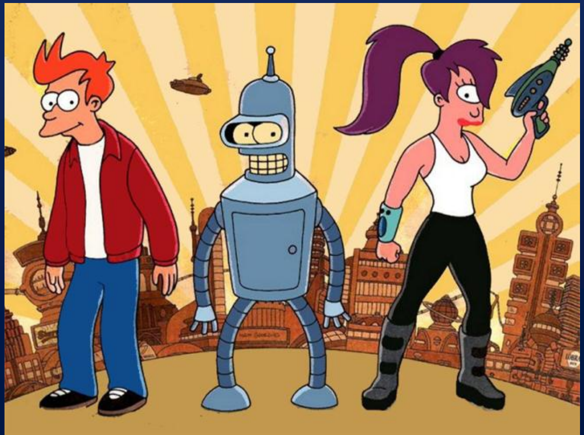
"Traditional" Animated Features…

Some examples:

• Automating Key-framing in many Disney-type animations

• The flocking behaviour of the wild-beast in Lion King

• Non photorealistic rendering: 3D effects in Futurama



Movies

Behind the scenes on *Antz* Production

Number of frames in the movie	119,592
Number of times the movie was rendered during production	**15 (approx.)**
Number of feet of approved animation produced in a week	107 ft.
Total number of hours of rendering per week	**275,000 hrs.**
Average size of the frame rendered	6 MB

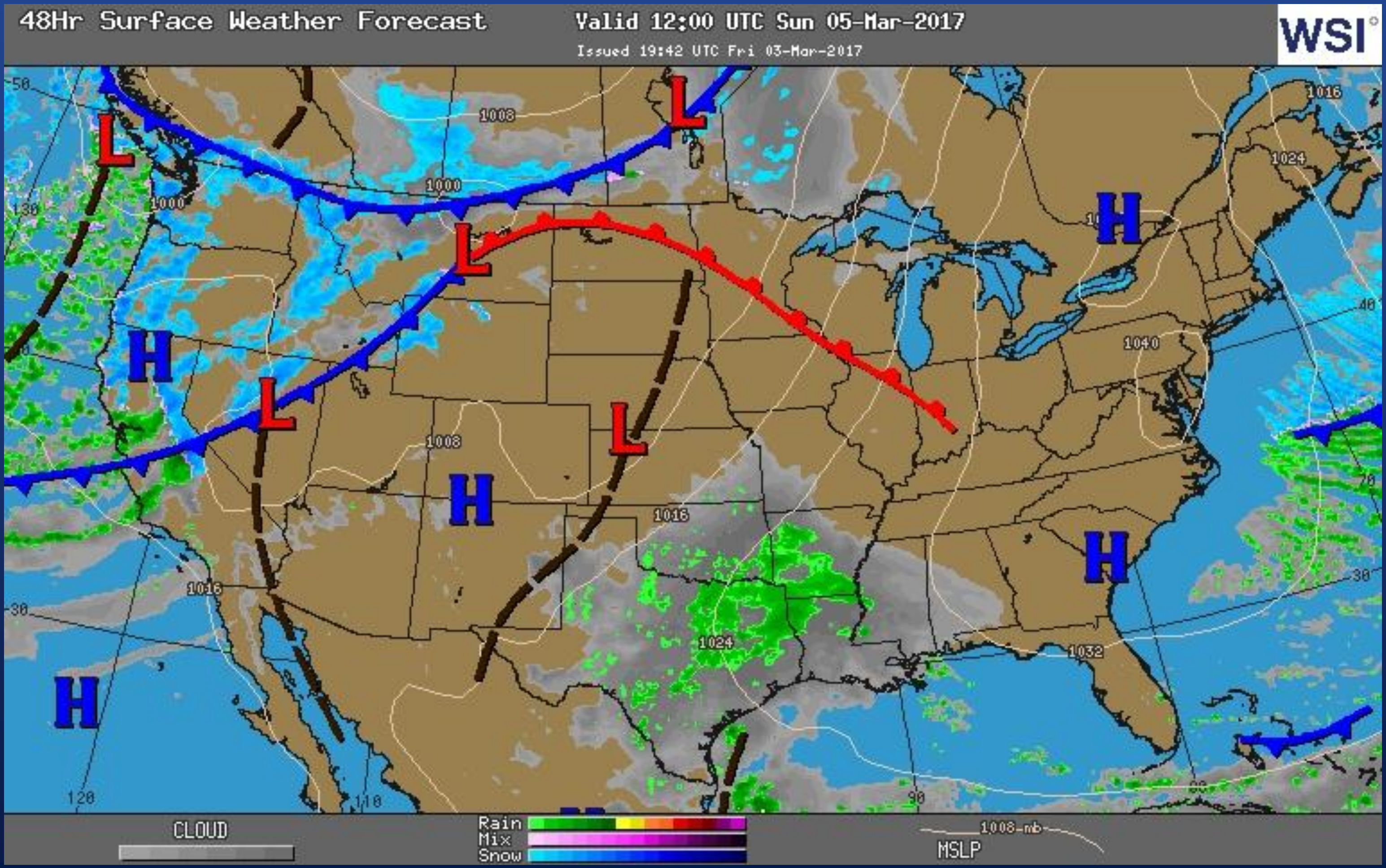
Behind the scenes on *Antz* Production

Total number of Silicon Graphics servers used for rendering	**270**
Number of desktop systems used in production	**166**
Total Number of processors used for rendering	**700**
Average amount of memory per processor	**256 MB**

Simulations

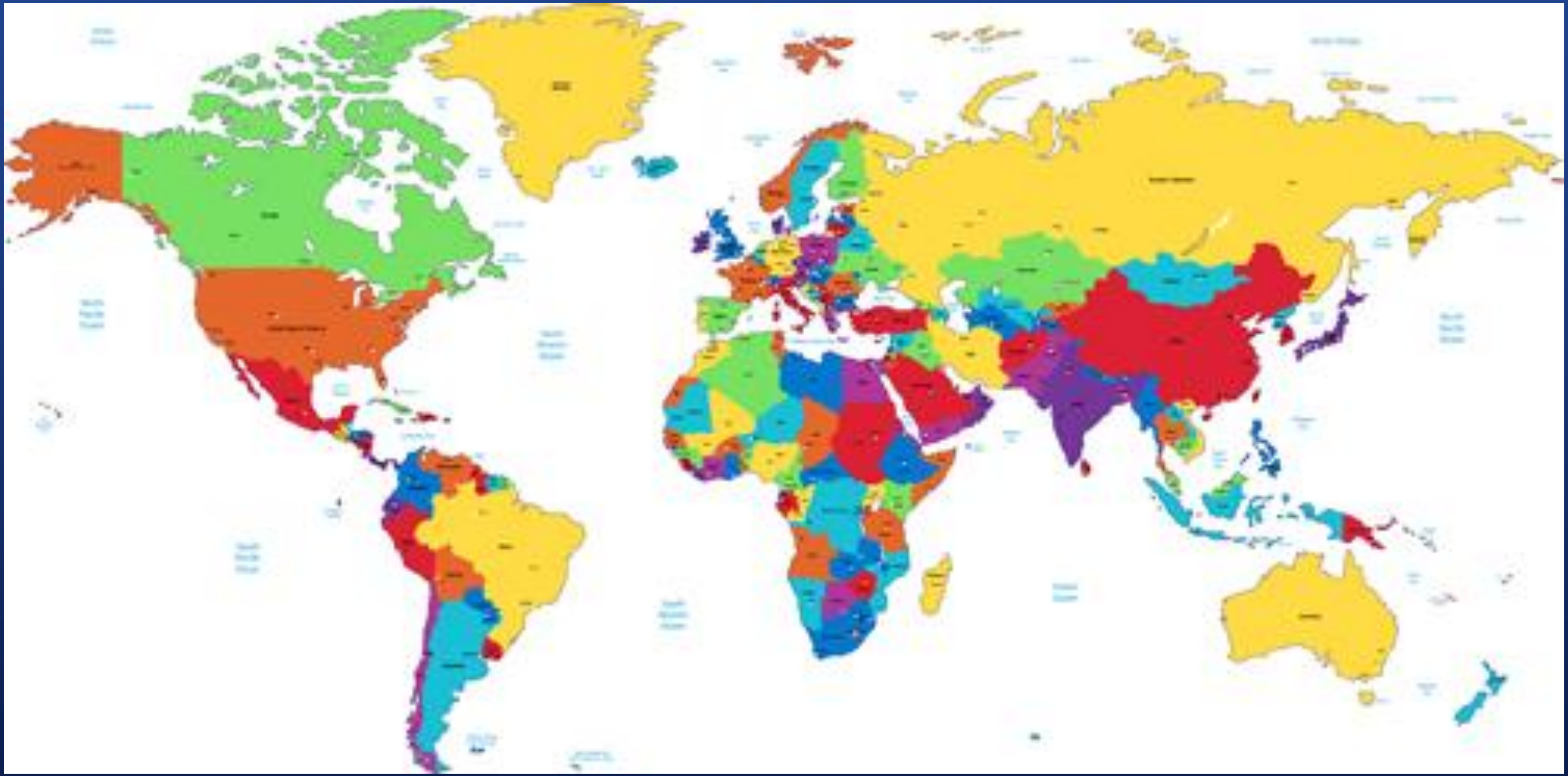
Game

Weather Map

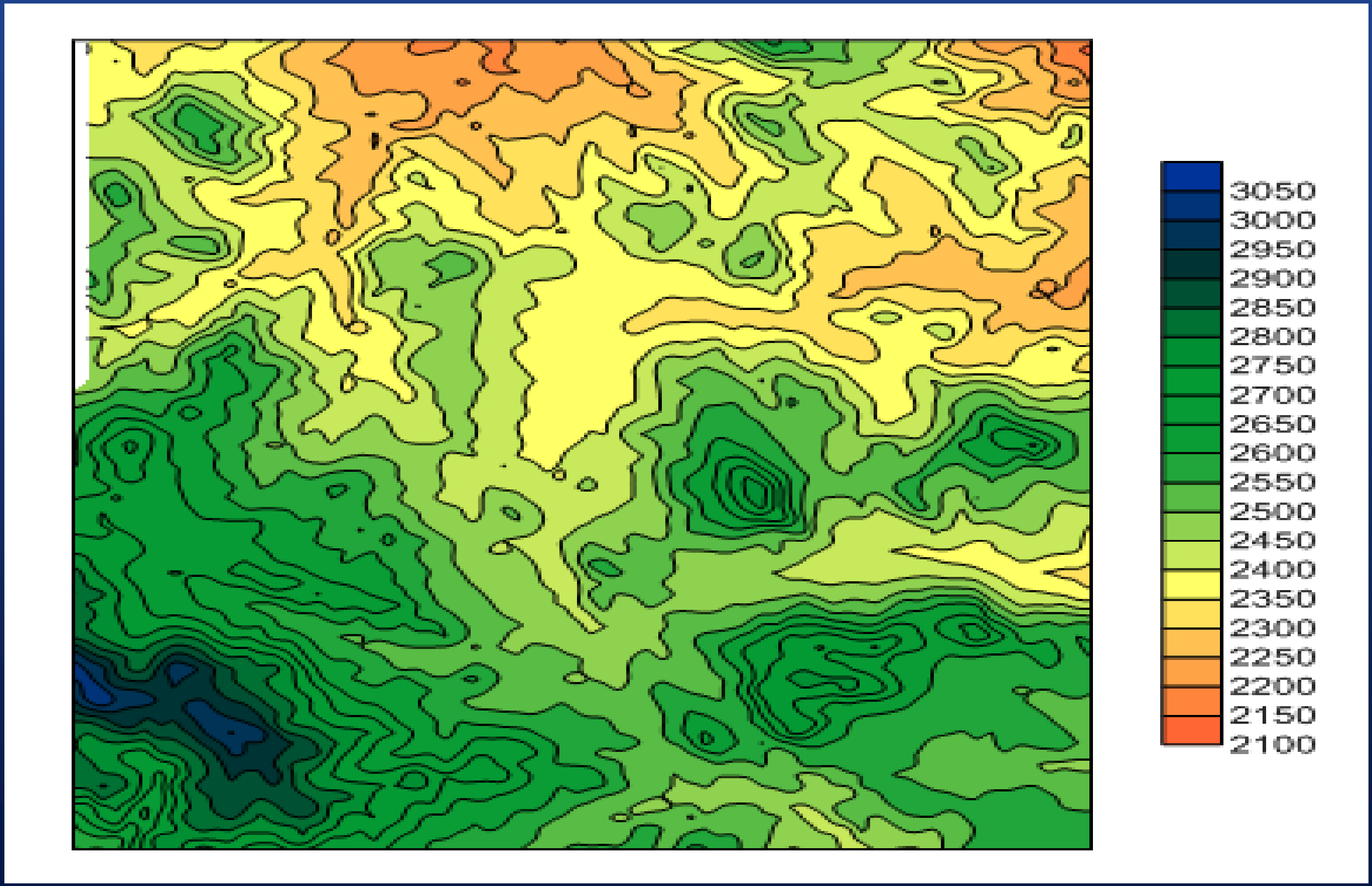


48Hr Surface Weather Forecast
Valid 12:00 UTC Sun 05-Mar-2017
Issued 19:42 UTC Fri 03-Mar-2017
WSI
CLOUD
Rain
Mix
Snow
1008 mb
MSLP

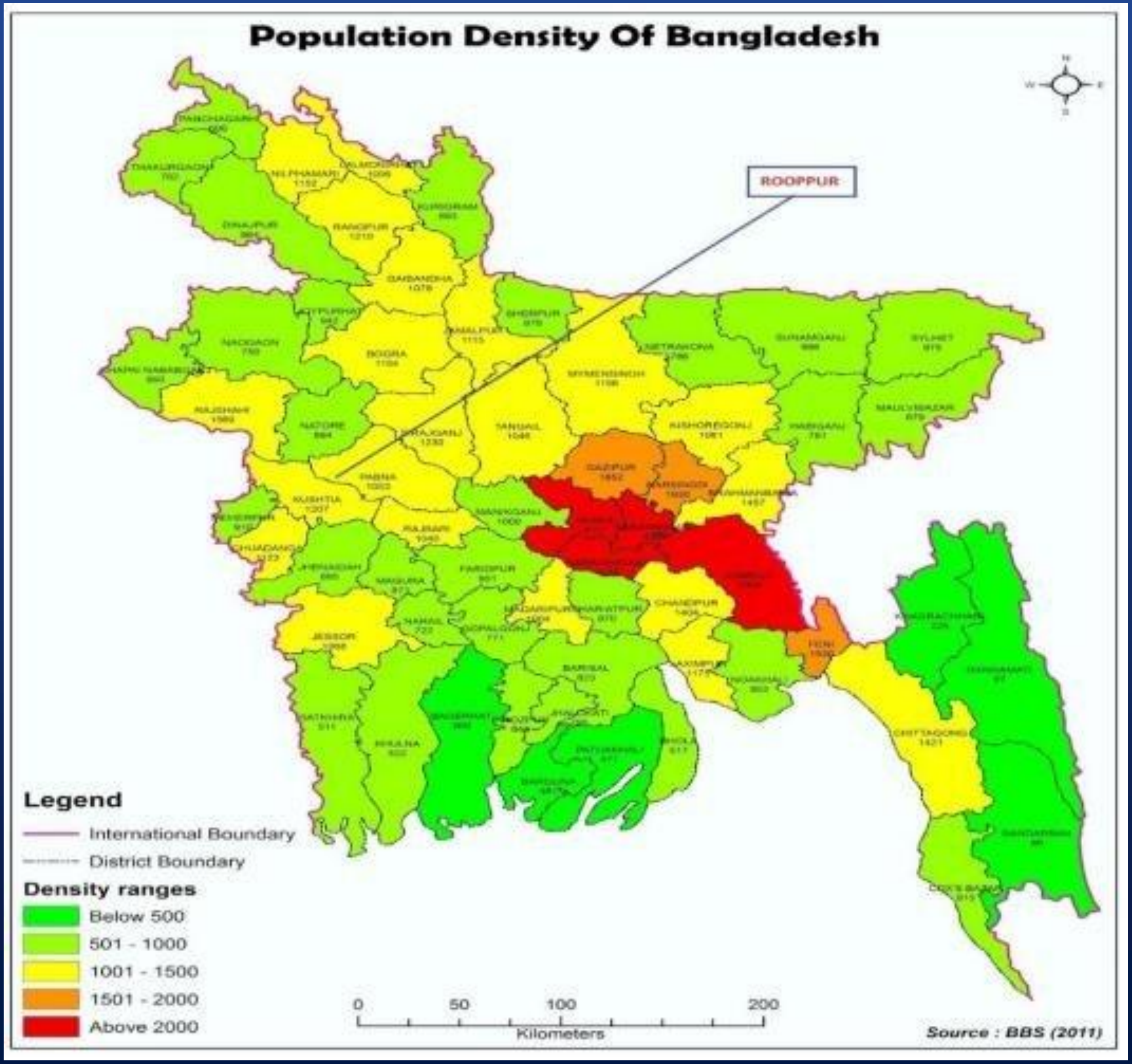
Geographical Map



Contour Map


3050
3000
2950
2900
2850
2800
2750
2700
2650
2600
2550
2500
2450
2400
2350
2300
2250
2200
2150
2100

Population Density Map



Computer Graphics Systems

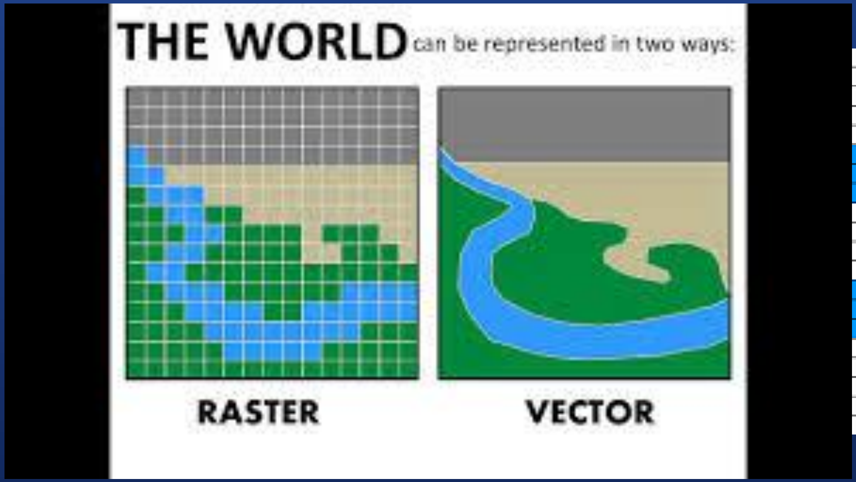
- Active
- Passive

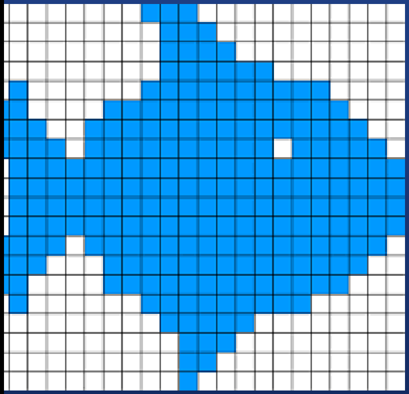
In both the cases, the input to the system is the scene description and output is a static or animated scene to be displayed.

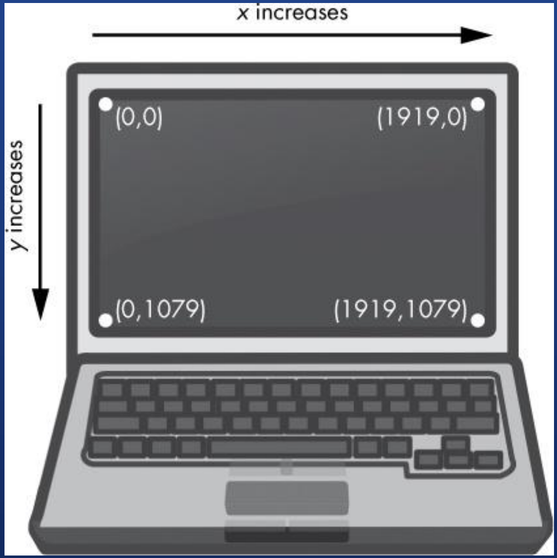
In case of active systems, the user controls the display with the help of a GUI, using an input device.

Image Representation

A digital image is composed of discrete pixels. The pixels are arranged in a row and column fashion to form a rectangular area, called as a raster.



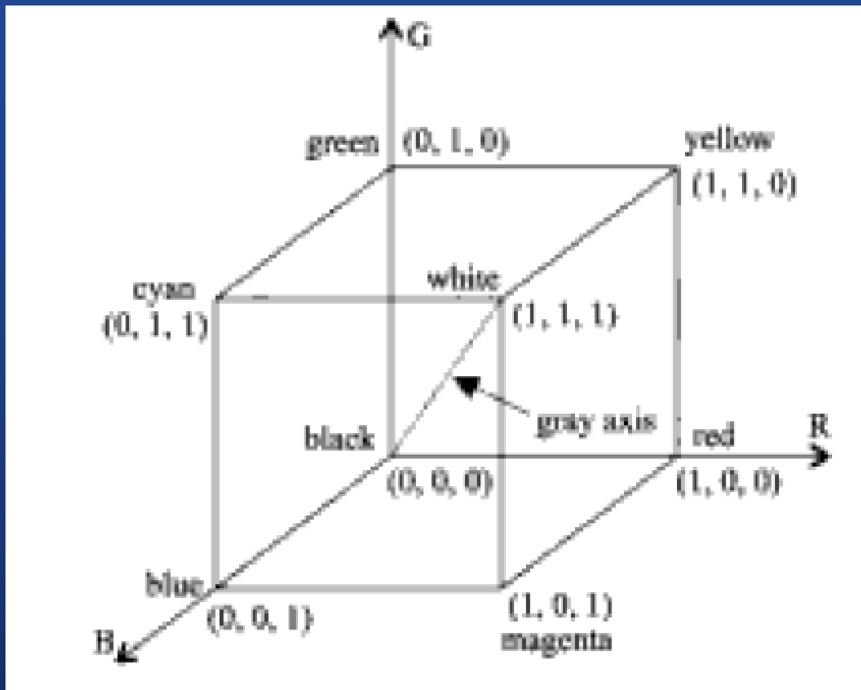


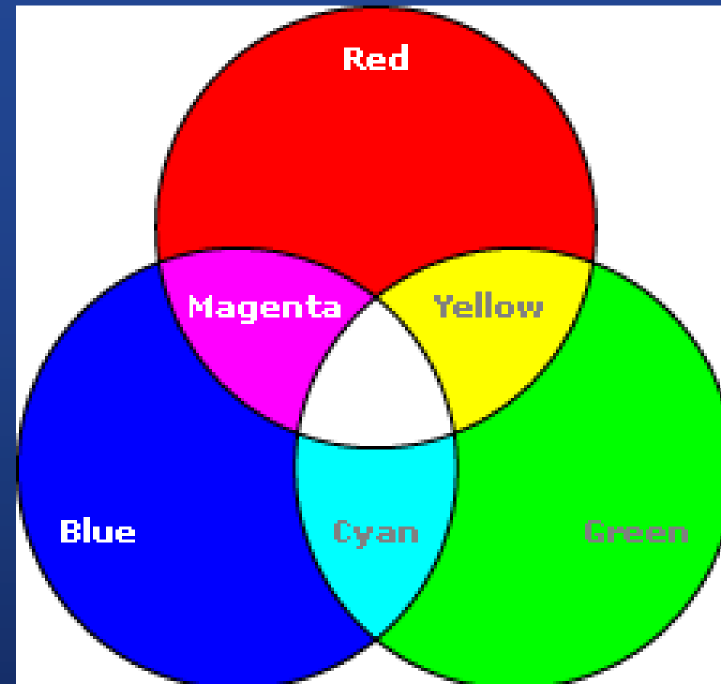


A 3x2 inch image at a resolution of 300 pixels per inch would have a total of 540,000 pixels.

The coordinates of upper-left corner of a monitor is (0,0)

RGB Color Model

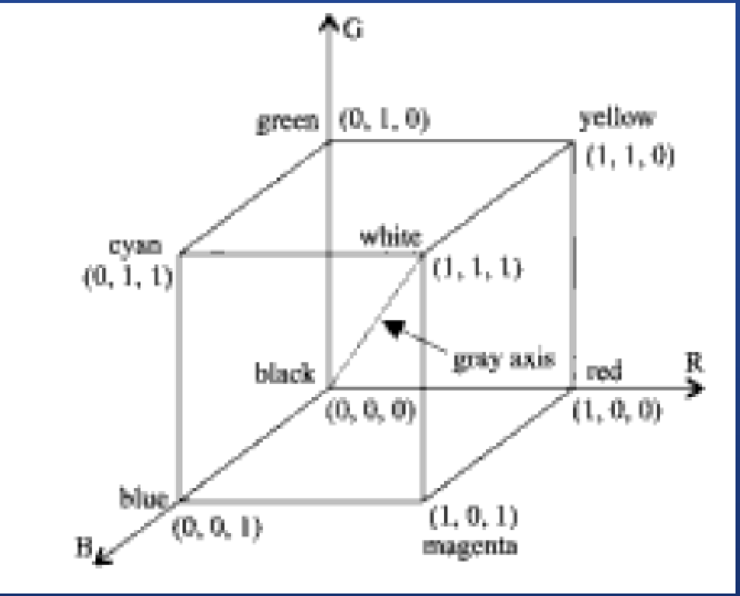




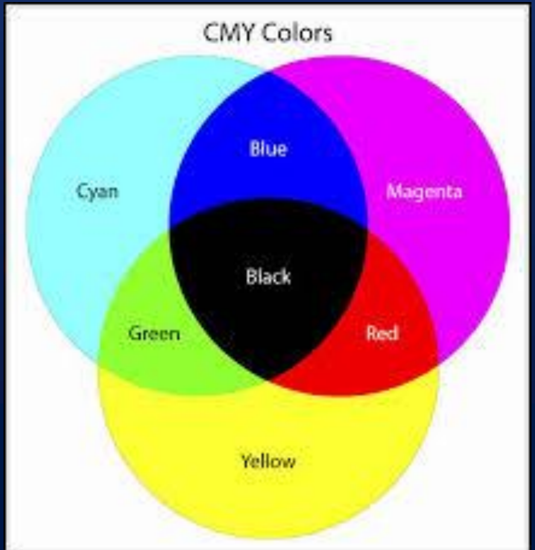
Given this RGB color model an arbitrary color within the cubic color space can be specified by its color coordinates: (r, g, b). For example, we have $(0, 0, 0)$ for black, $(1, 1, 1)$ for white, $(1, 1, 0)$ for yellow, etc. A gray color at $(0.7, 0.7, 0.7)$ has an intensity halfway between one at $(0.9, 0.9, 0.9)$ and one at $(0.5, 0.5, 0.5)$.

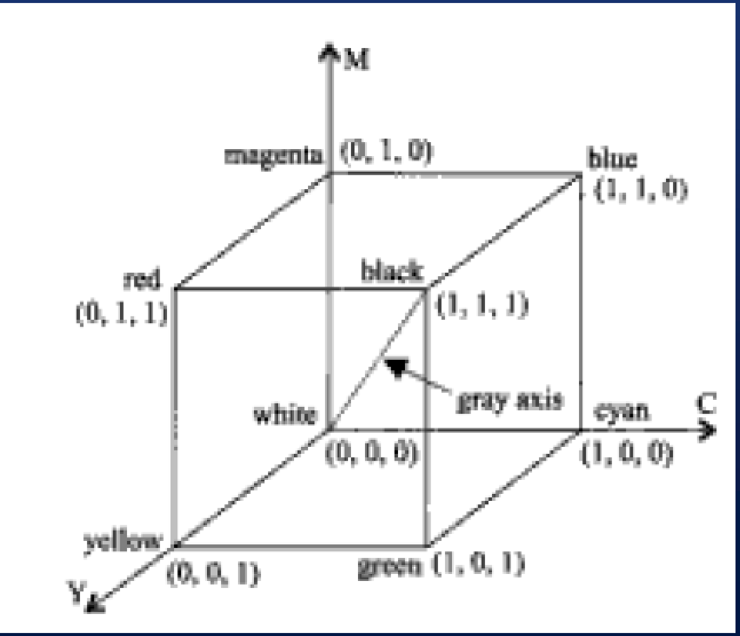
CMYK Color Model

The CMYK is a complementary color model, defines colors using subtracting process matches the principle of printer



$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} C \\ M \\ Y \end{pmatrix} \qquad \begin{pmatrix} C \\ M \\ Y \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$





Color Coding

Image representation is essentially the representation of pixel colors. Using direct coding we allocate a certain amount of storage space for each pixel to code its color. For example, we may allocate 3 bits for each pixel, with one bit for each primary color (see Fig. 2-3). This 3-bit representation allows each primary to vary independently between two intensity levels: 0 (off) or 1 (on). Hence each pixel can take on one of the eight colors that correspond to the corners of the RGB color cube.

bit 1: r	bit 2: g	bit 3: b	color name
0	0	0	black
0	0	1	blue
0	1	0	green
0	1	1	cyan
1	0	0	red
1	0	1	magenta
1	1	0	yellow
1	1	1	white

Fig. 2-3 Direct coding of colors using 3 bits.

A widely accepted industry standard uses 3 bytes, or 24 bits, per pixel, with one byte for each primary color. This way we allow each primary color to have 256 different intensity levels, corresponding to binary values from 00000000 to 11111111. Thus a pixel can take on a color from $256 \times 256 \times 256$ or 16.7 million possible choices. This 24-bit format is commonly referred to as the true color representation, for the difference between two colors that differ by one intensity level in one or more of the primaries is virtually undetectable under normal viewing conditions. Hence a more precise representation involving more bits is of little use in terms of perceived color accuracy.

Color Coding

A notable special case of direct coding is the representation of black-and-white (bilevel) and gray-scale images, where the three primaries always have the same value and hence need not be coded separately. A black-and-white image requires only one bit per pixel, with bit value 0 representing black and 1 representing white. A gray-scale image is typically coded with 8 bits per pixel to allow a total of 256 intensity or gray levels.

Although this direct coding method features simplicity and has supported a variety of applications, we can see a relatively high demand for storage space when it comes to the 24-bit standard. For example, a 1000×1000 true color image would take up three million bytes. Furthermore, even if every pixel in that image had a different color, there would only be one million colors in the image. In many applications the number of colors that appear in any one particular image is much less. Therefore the 24-bit representation's ability to have 16.7 million different colors appear simultaneously in a single image seems to be somewhat overkill.

End of Chapter 1