

NCS-403: Computational Structures in Computer Graphics & Types

Overview of Graphics Systems

Unit 1 – Lecture - 1

Outline

- Survey of Computer Graphics
- Overview of Graphics Systems
 - Image Basics
 - Graphics Hardware
 - Input
 - Describing something to the computer
 - Computation
 - Computing what we want to draw
 - Output
 - Final representation

What is the goal of computer graphics?

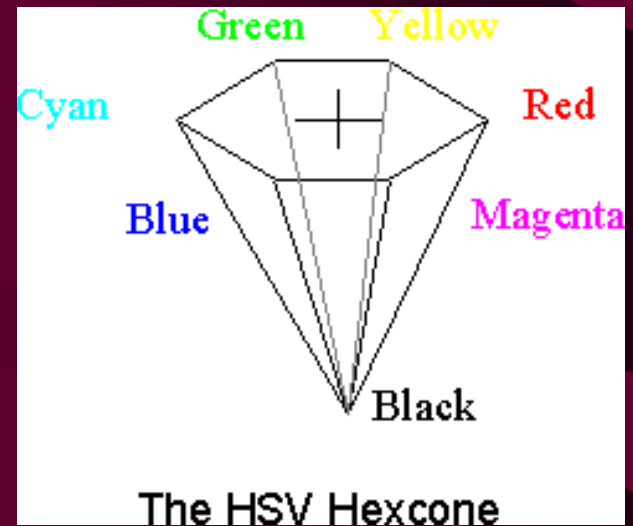
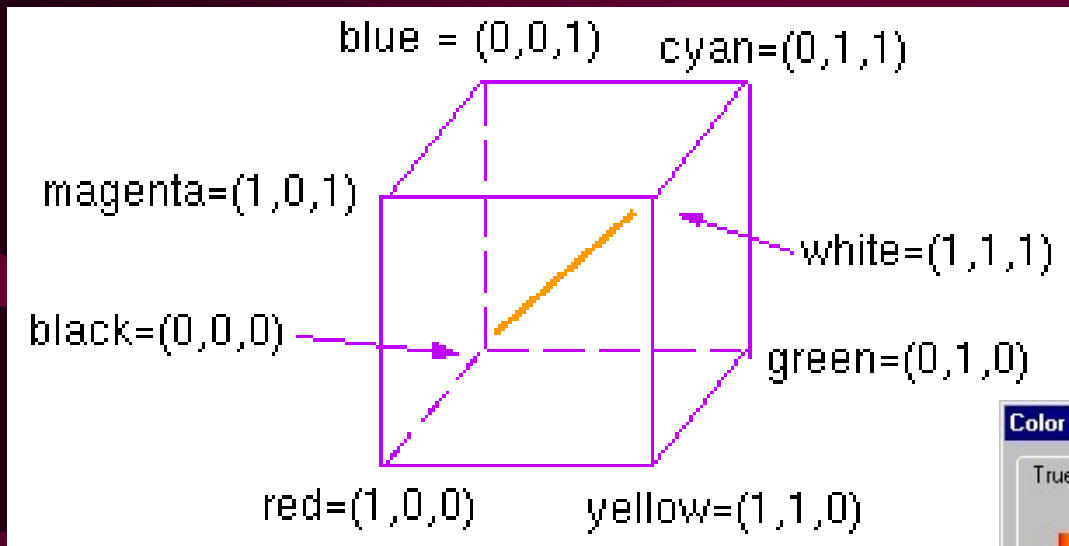
- High level, why computer graphics?
- Lower level, what is the computer doing?

Business of Generating Images

- Images are made up of pixels

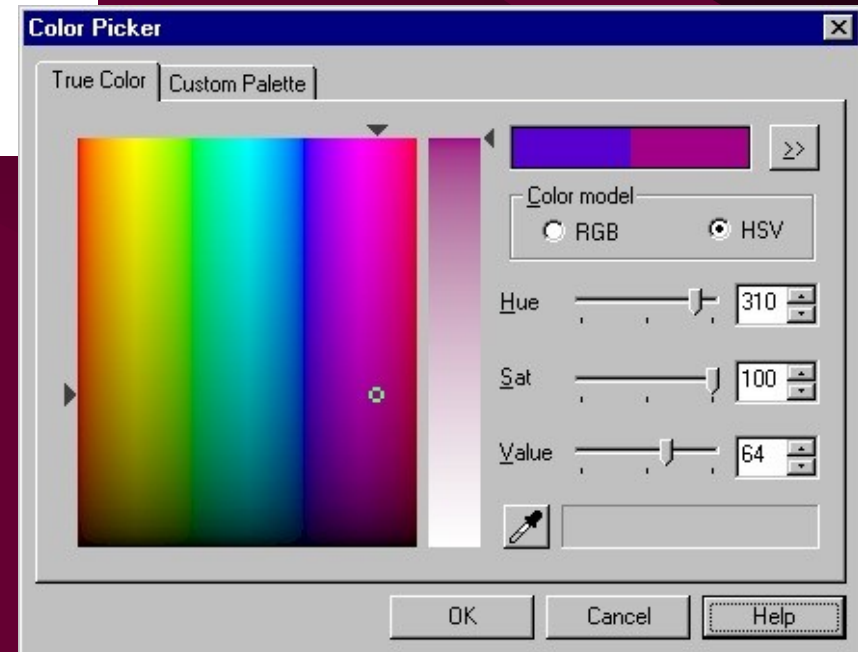


RGB



RGB Color cube (what we use in computer graphics)

Other color spaces include HSV, YUV, YCrCb, and YIQ



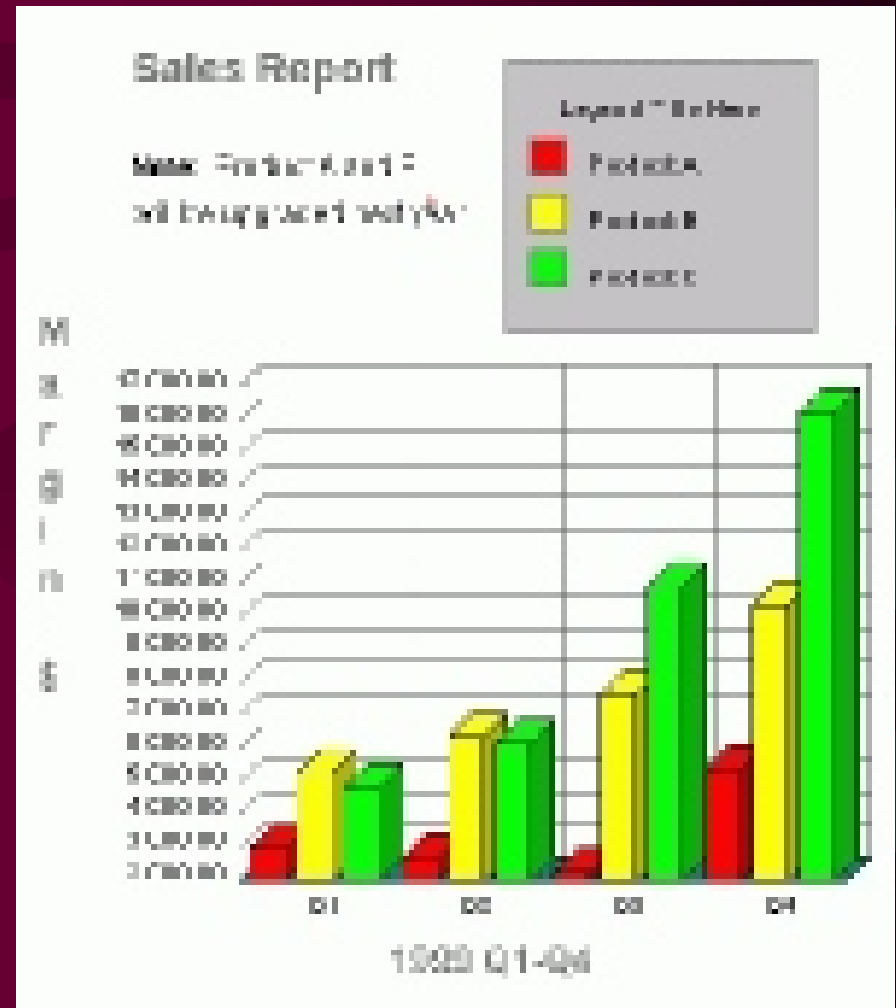
The “goal” of computer graphics

- Solve the function
 - Red @ a pixel is $f(i,j)=\dots$
 - Green @ a pixel is $f(i,j)=\dots$
 - Blue @ a pixel is $f(i,j)=\dots$



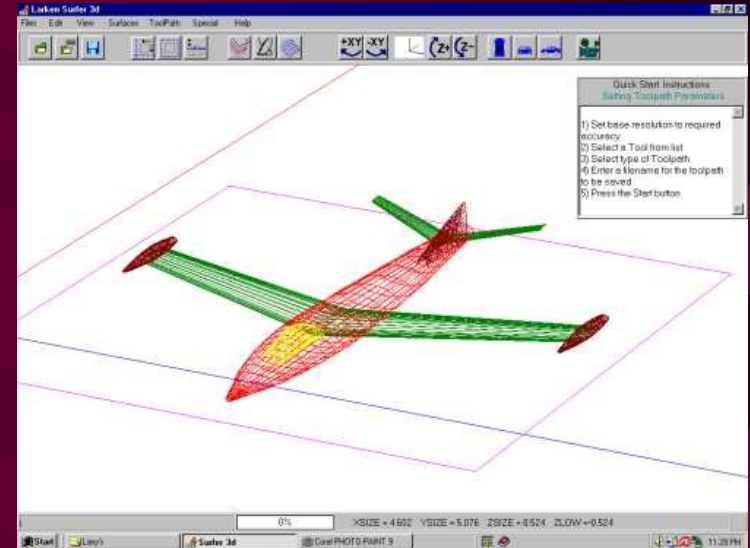
Early Applications of Computer Graphics

- Data Visualization
 - Charts and Graphs



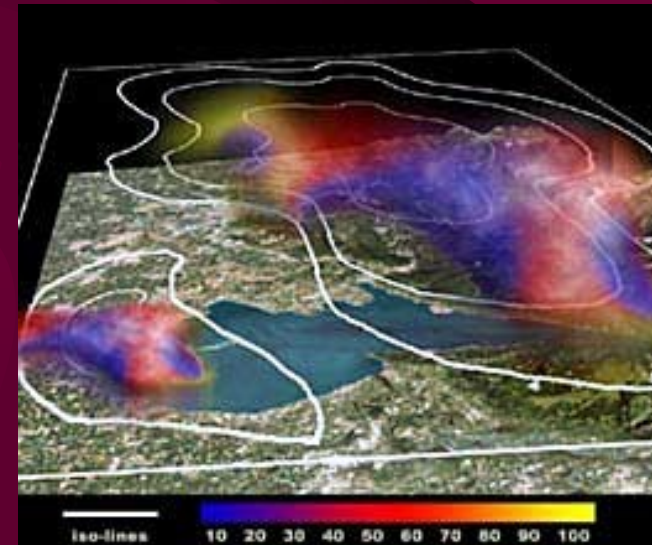
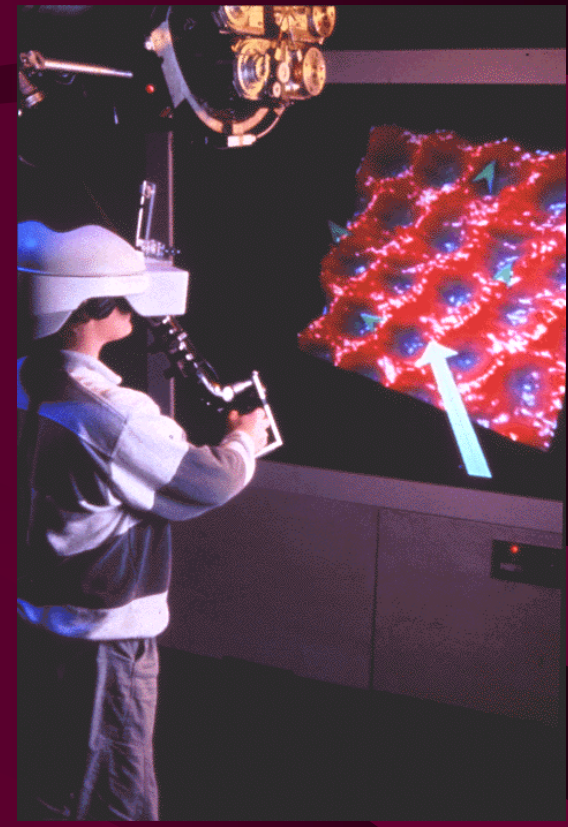
Early Applications of Computer Graphics

- Computer Aided Design (CAD)
 - Q: Why wireframe?
- Why these apps?
 - A: Better conceptualization, interaction, transfer of ideas



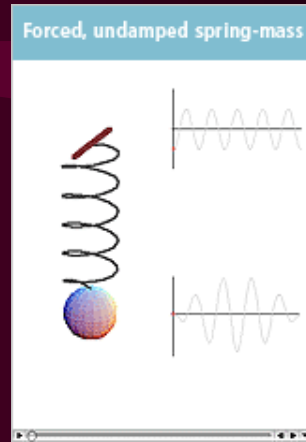
Computer Graphics Applications

- Virtual Reality
 - VR: User interacts and views with a 3D world using “more natural” means
 - Best VR?
- Data Visualization
 - Scientific, Engineering, Medical data
 - Visualizing millions to billions of data points
 - See trends
 - Different schemes



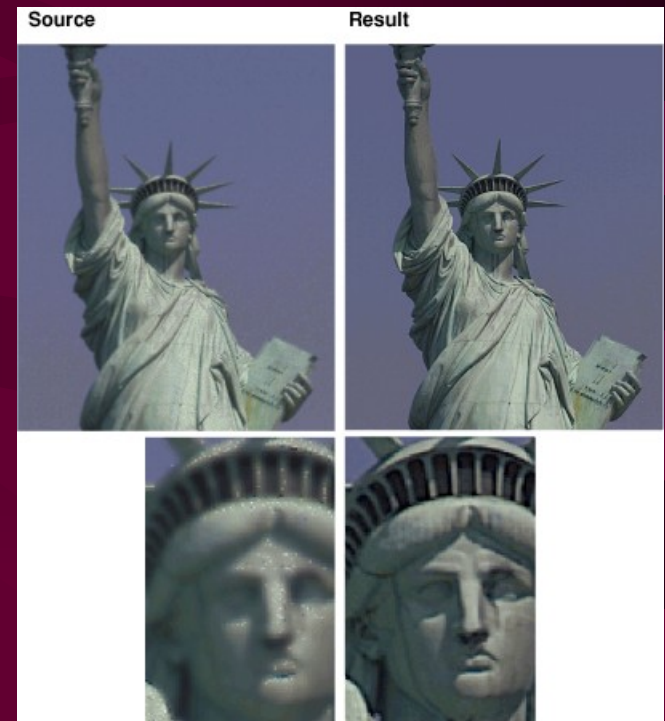
Computer Graphics Applications

- Education and Training
 - Models of physical, financial, social systems
 - Comprehension of complex systems
- Computer Art
 - Fine and commercial art
 - Performance Art
 - Aesthetic Computing
 - SIGGRAPH
- Games/Movies



Computer Graphics Applications

- Image Processing
 - ~Inverse of Graphics
 - Start with a picture
 - Process picture information
- Graphical User Interfaces (GUIs)
 - WIMP interface
 - HCI

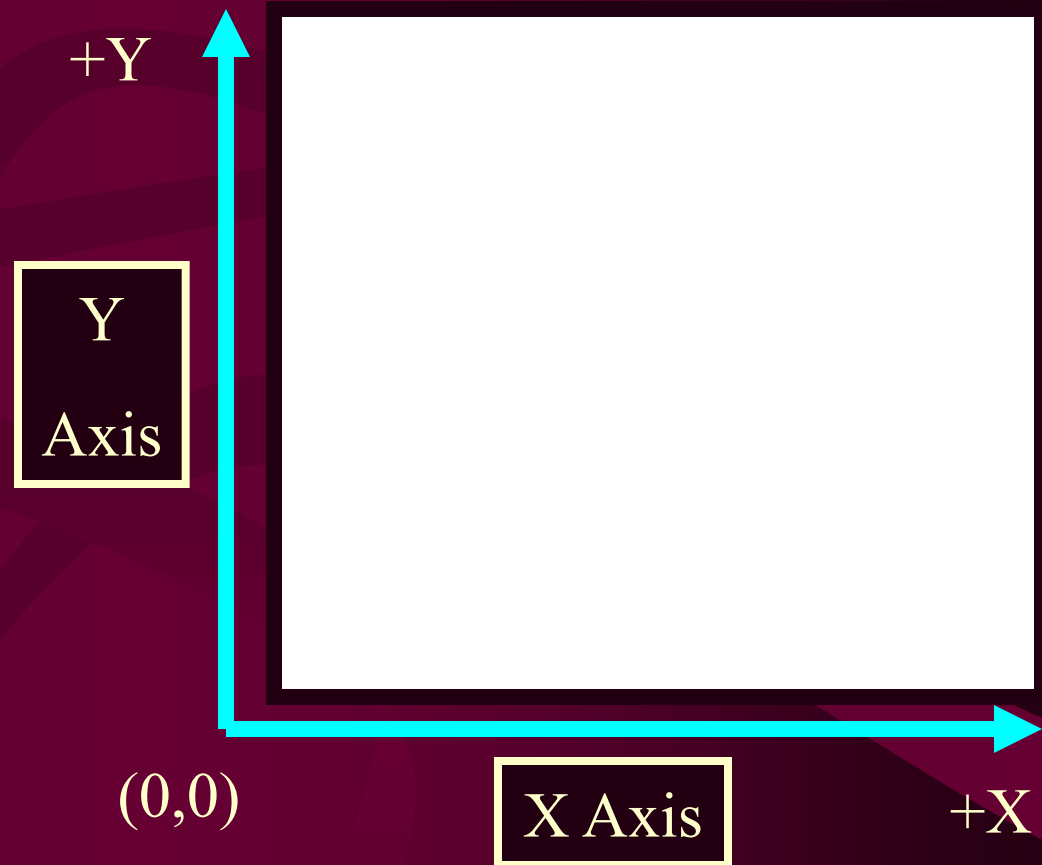


Overview of Graphics Systems

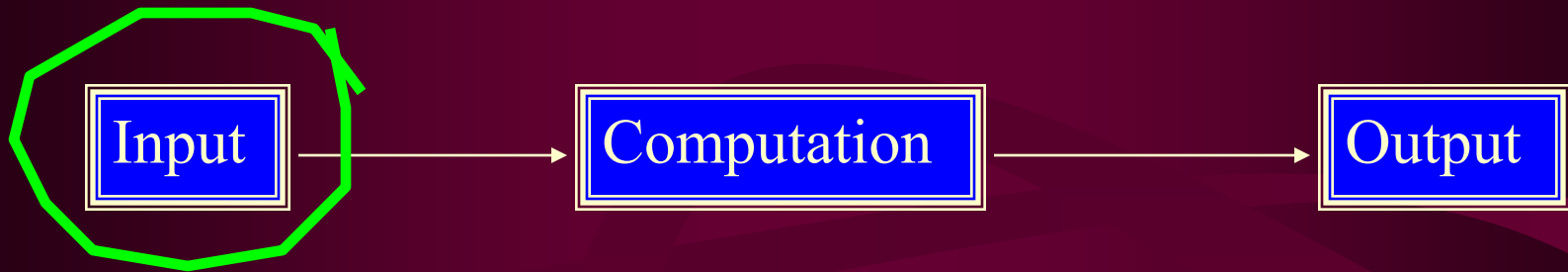
- Images
- Hardware
 - Input Systems
 - Output Systems
- Software
 - OpenGL

Two Dimensional Images

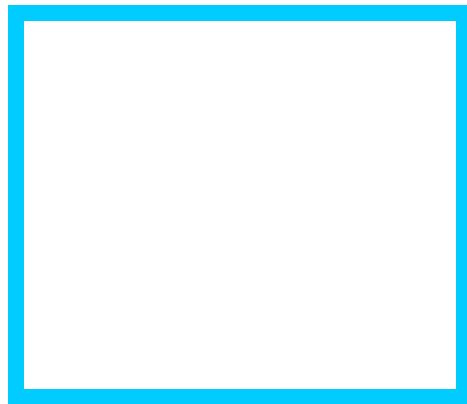
- Images (at least the ones in this class) are two dimensional shapes.
- The two axes we will label as X (horizontal), and Y (vertical).



Hardware Pipeline



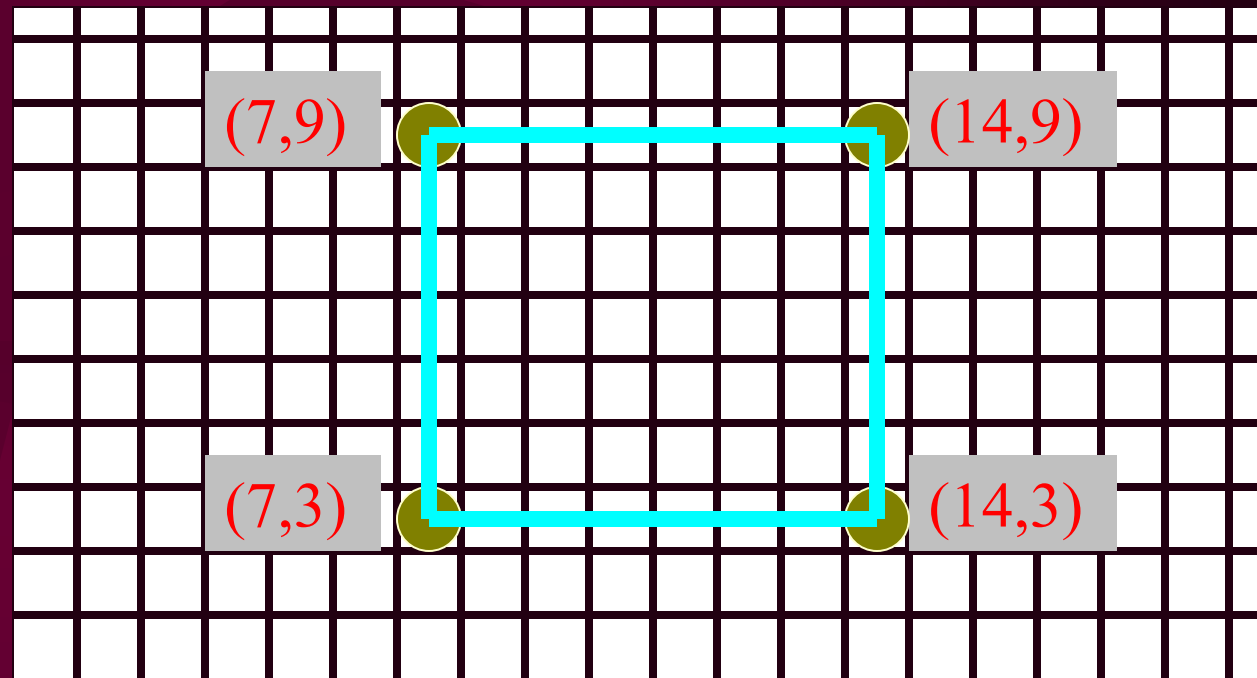
We want to draw a rectangle, how do we describe it to a computer?



Model (n) - object description that a computer understands.

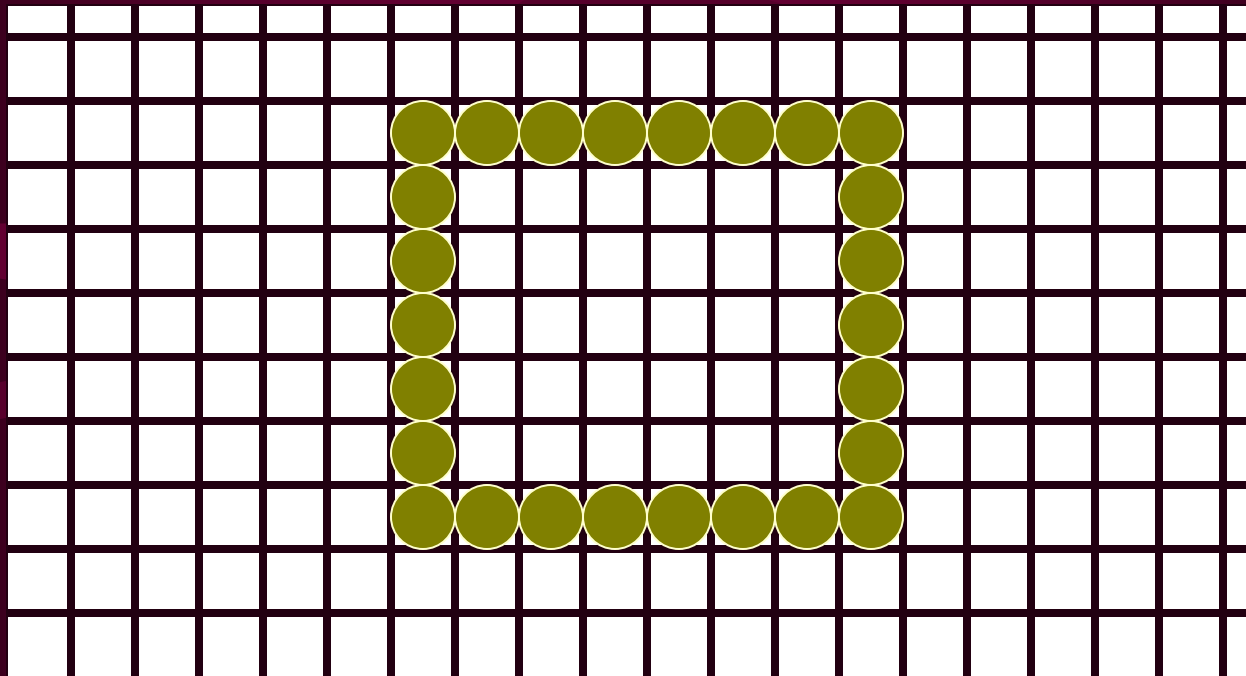
Partition the space

1. Define a set of points (vertices) in 2D space.
2. Given a set of vertices, draw lines between consecutive vertices.



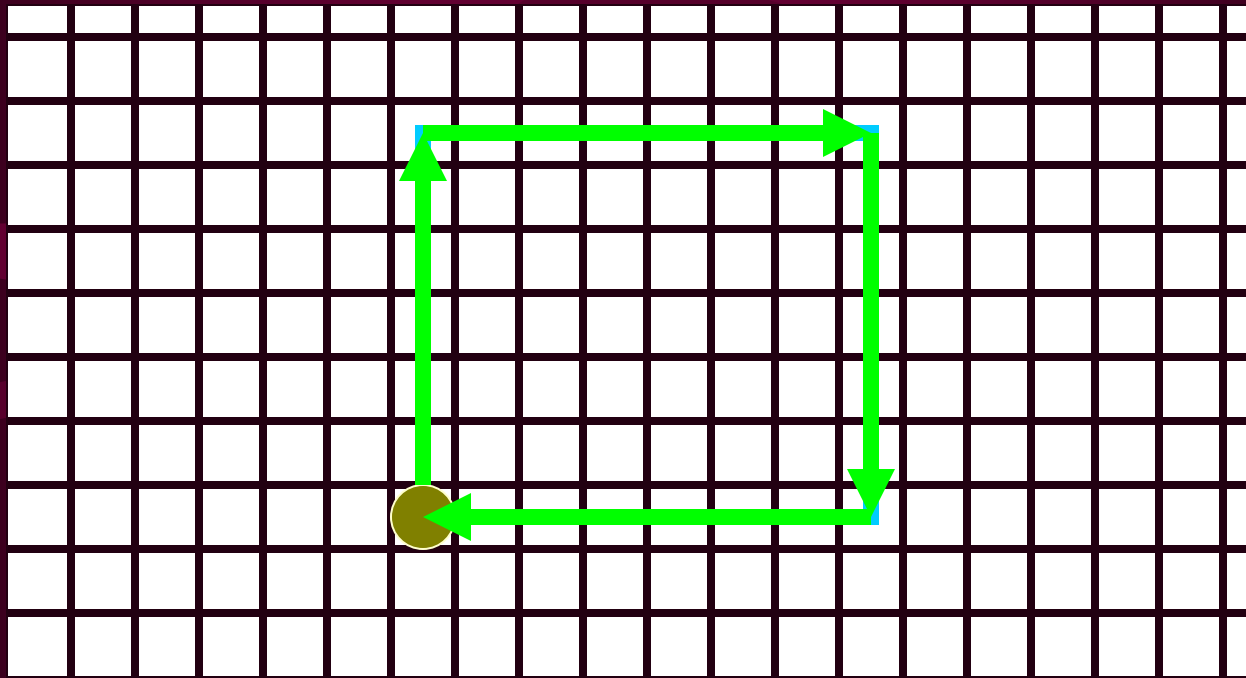
Vertex (pl. Vertices) - a point in 2 or 3 dimensional space.

Record every position



Bitmap - a rectangular array of bits mapped one-to-one with pixels.

Position relative



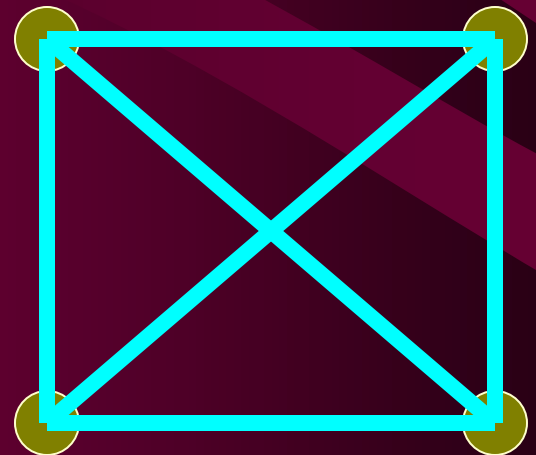
Vector display system - graphical output system that was based on strokes (as opposed to pixels). Also known as: *random*, *calligraphic*, or *stroke* displays.

Representing Objects

- Most common method is the VERTEX method. Define the object as a set of points with connectivity information.
- Why is connectivity important?

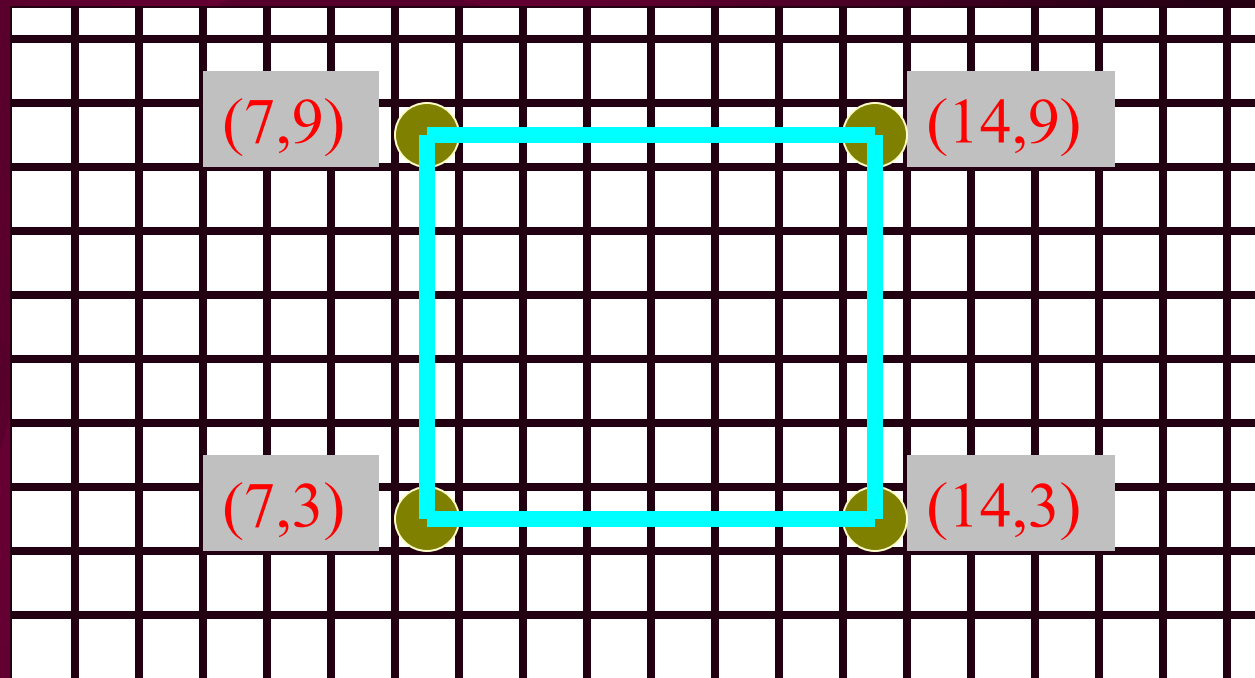
Connectivity - information that defines which vertices are connected to which other vertices via edges.

Edge - connects two vertices



Model file for rectangle

- v 4 e 4
- 7 3
- 7 9
- 14 9
- 14 3
- 1 2
- 2 3
- 3 4
- 4 1



How do we do this?



Input Devices

- Locator Devices
- Keyboard
- Scanner
 - Images
 - Laser
- Cameras (research)

Locator Devices

When queried, locator devices return a position and/or orientation.

- Mouse (2D and 3D)
- Trackball
- Joystick (2D and 3D)



Locator Devices

When queried, locator devices return a position and/or orientation.

- Tablet
- Virtual Reality Trackers
 - Data Gloves
 - Digitizers

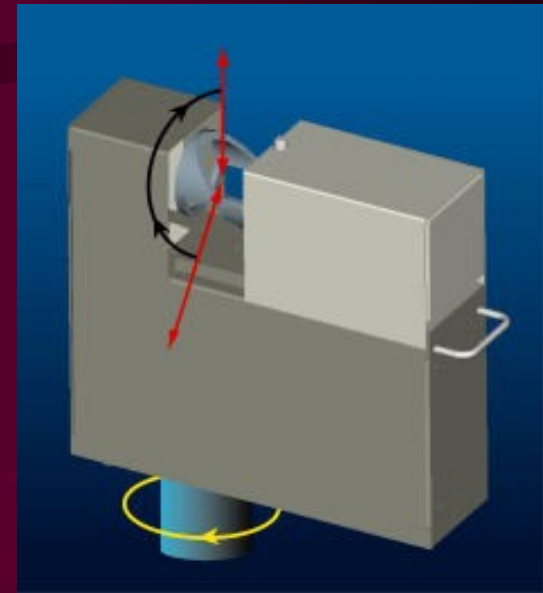


Keyboard

- Text input
 - List boxes, GUI
 - CAD/CAM
 - Modeling
- Hard coded
 - Vertex locations are inserted into code

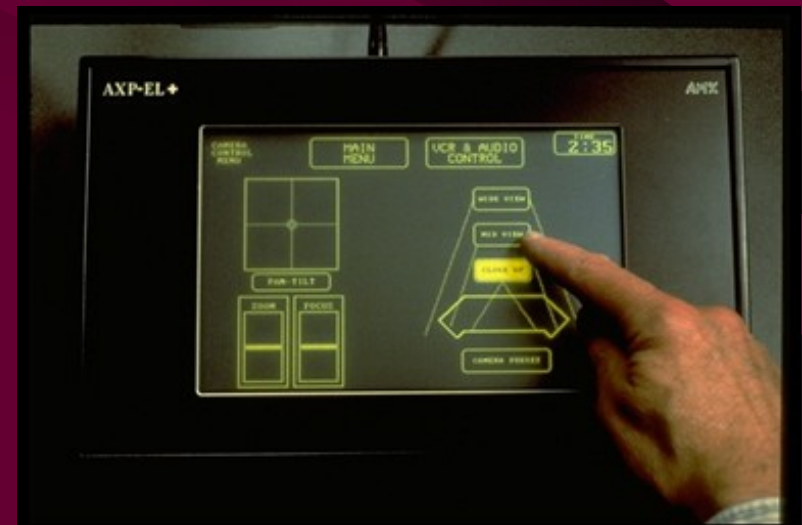
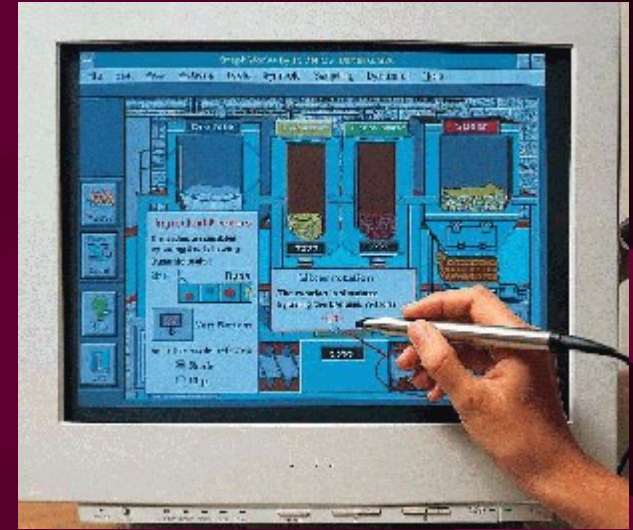
Scanners

- Image Scanners - Flatbed, etc.
 - What type of data is returned? Bitmap
- Laser Scanners - Deltasphere
 - Emits a laser and does time of flight. Returns 3D point
- Camera based - research
 - Examine camera image(s) and try to figure out vertices from them.



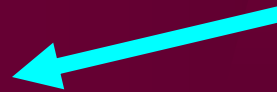
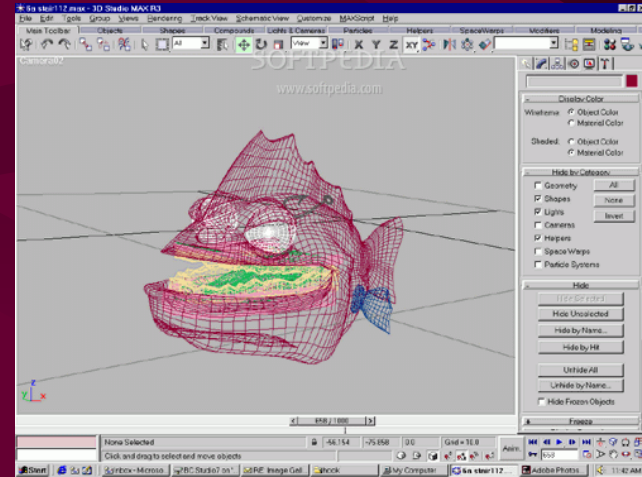
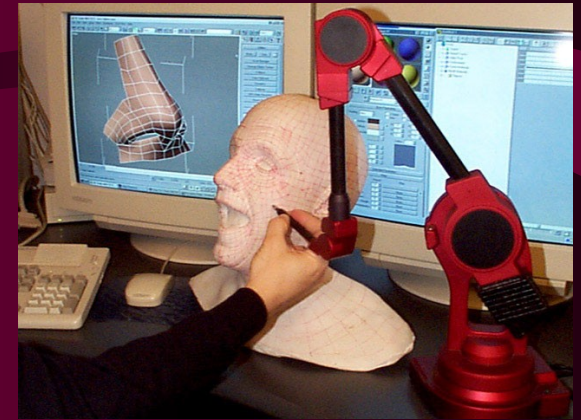
Many others

- Light Pens
- Voice Systems
- Touch Panels
- Camera/Vision Based
- Which is best?



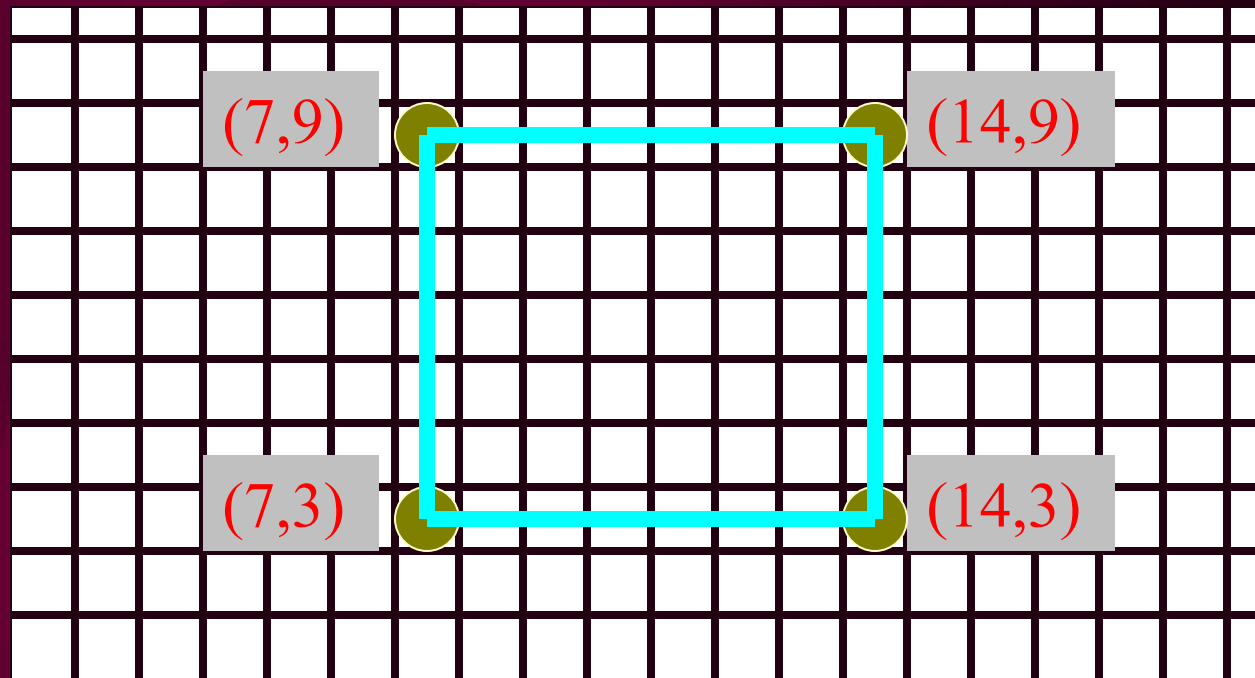
Common Modeling Approach

- Hybrid
- Animator jobs

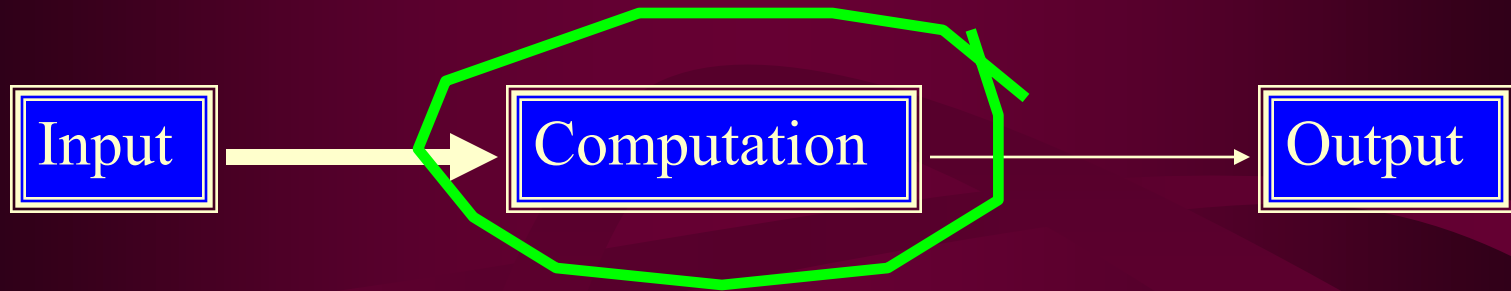


Model file for rectangle

- v 4 e 4
- 7 3
- 7 9
- 14 9
- 14 3
- 1 2
- 2 3
- 3 4
- 4 1



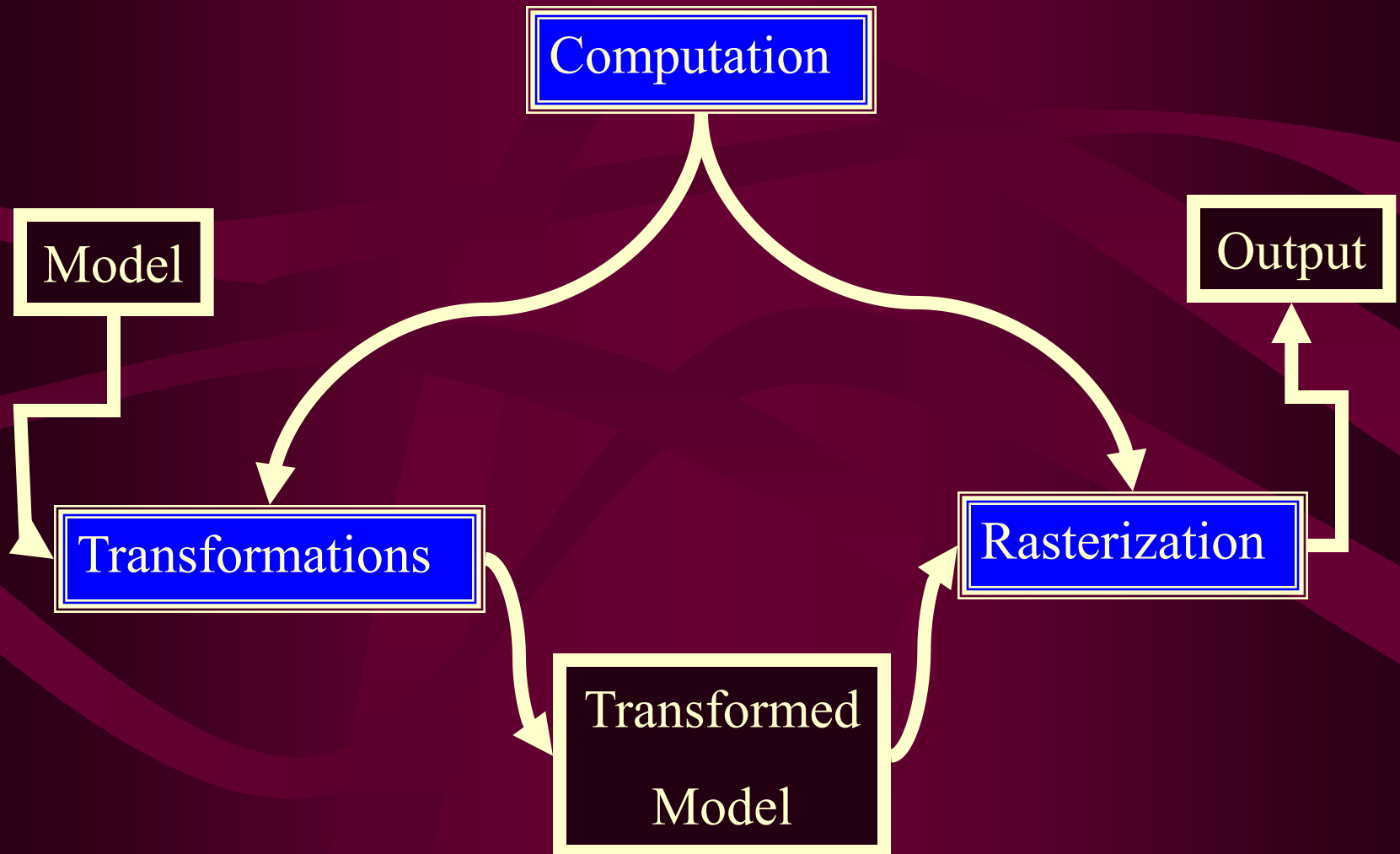
Computation Stage



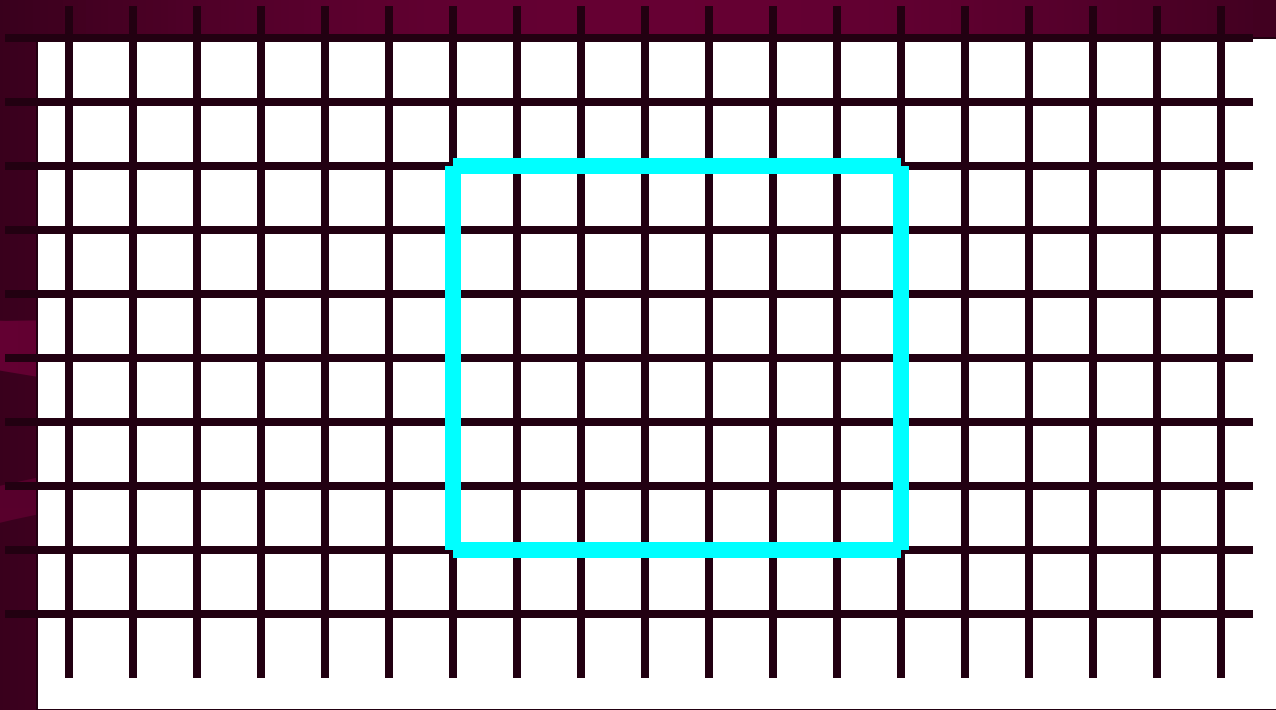
- Now that we have a *model* of what we want to draw, what goes on inside the computer to generate the output?



Computation Stage



How do we store this?



We would like to allocate memory to hold the results of the computation stage.

Framebuffer

Framebuffer - A block of memory, dedicated to graphics output, that holds the contents of what will be displayed.

Pixel - one element of the framebuffer

Framebuffer

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Questions:

What is the minimum number of bytes for the framebuffer?

Framebuffer in Memory

- If we want a framebuffer of 640 pixels by 480 pixels, we should allocate:

$$\text{framebuffer} = 640 * 480 \text{ bits}$$

- How many bit should we allocate?

Q: What do more bits get you?

A: More values to be stored at each pixel.

Why would you want to store something other than a 1 or 0?

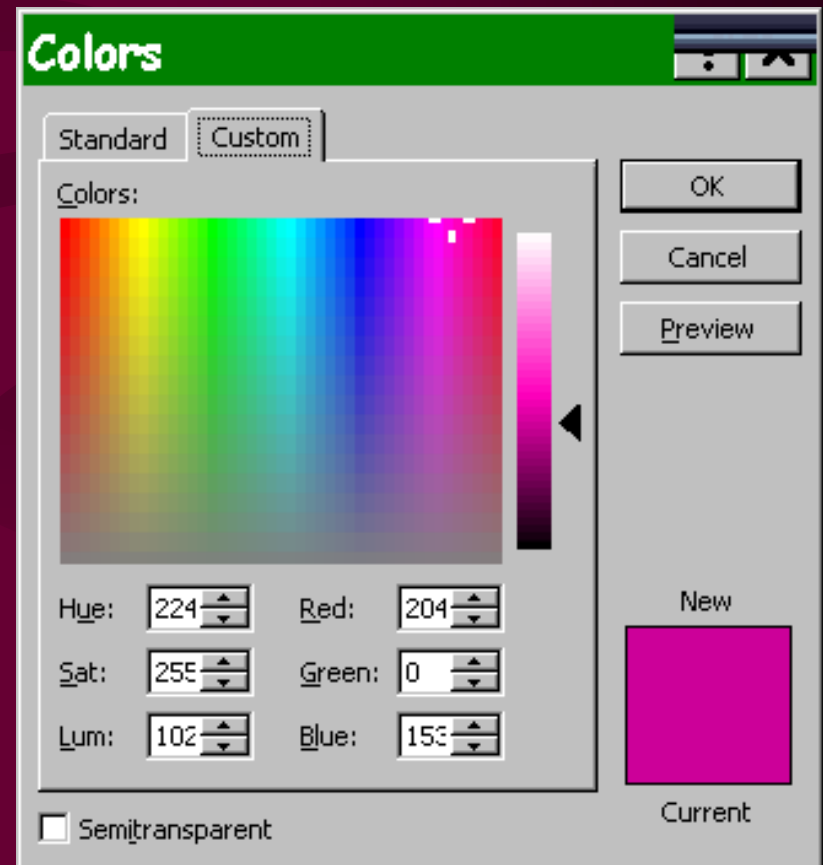
Framebuffer bit depth

- How many colors does 1 bit get you?
- How many colors do 8 bits get you?
 - Monochrome systems use this (green/gray scale)
- What bit depth would you want for your framebuffer?

bit depth - number of bits allocated per pixel in a buffer

Framebuffer bit depths

- Remember, we are asking “how much memory do we allocate to store the color at each pixel?”
- Common answers:
 - 16 and 32 bits



Bit depths

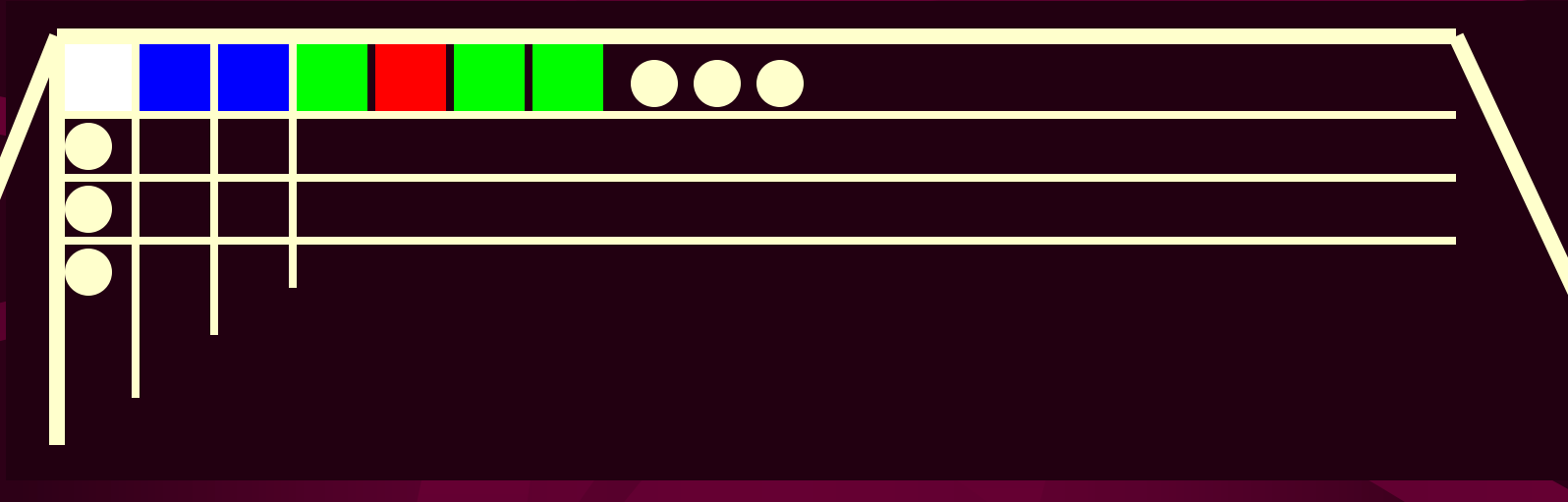
- 16 bits per pixel (high color)
 - 5 bits for red, 5/6 bits for green, 5 bits for blue
 - potential of 32 reds, 32/64 green, 32 blues
 - total colors: 65536
- 32 bits per pixel (true color)
 - 8 bits for red, green, blue, and alpha
 - potential for 256 reds, greens, and blues
 - total colors: 16777216 (more than the eye can distinguish)
- Let's look at Display Control Panel

Data Type Refresher

- bit - a 0 or 1. Can represent 2 unique values
- byte - 8 bits. 256 values
- word - 32 bits. 4,294,967,296 values
- int - 32 bits.
- float - 32 bits
- double - 64 bits
- unsigned byte - 8 bits

Memory

```
unsigned byte framebuffer[640*480*3];
```



```
framebuffer =
```

```
[255 255 255 0 0 255 0 0 255 0 255 0 255 0 0  
0 255 0 0 255 0 ...]
```

Graphic Card Memory

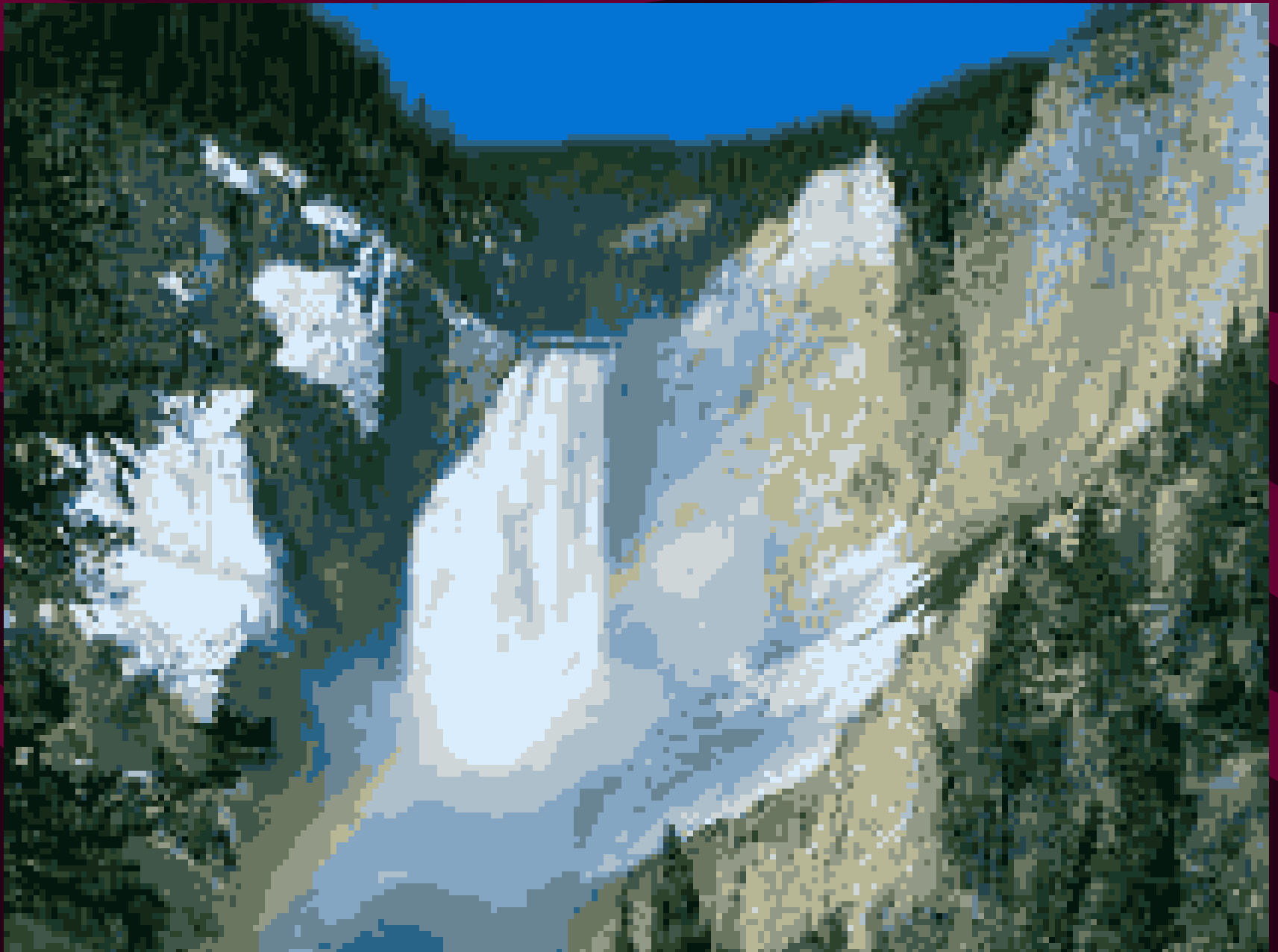
- How much memory is on our graphic card?
 - $640 * 480 * 32 \text{ bits} = 1,228,800 \text{ bytes}$
 - $1024 * 768 * 32 \text{ bits} = 3,145,728 \text{ bytes}$
 - $1600 * 1200 * 32 \text{ bits} = 7,680,000 \text{ bytes}$
- How much memory is on your graphics card?
- As a side note: Playstation 1 has 2 MB RAM. How do they do it? What is the TV resolution? 1 bit alpha, no z buffer.

A: Egads! Not enough memory!

Q: What is dithering?

- Trading spatial resolution for intensity and color depth.
- Sometimes call *digital half-toning*
- Increases the number of *apparent* colors than are actually capable of being displayed

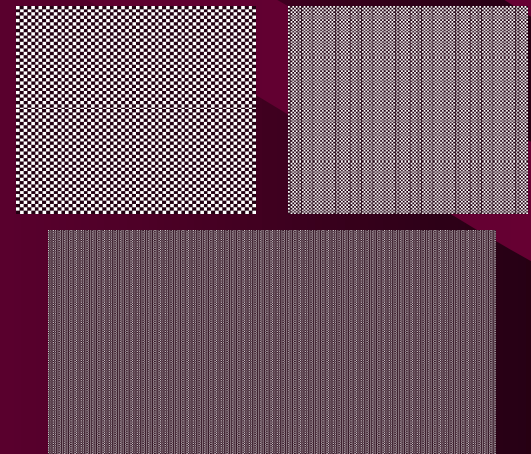
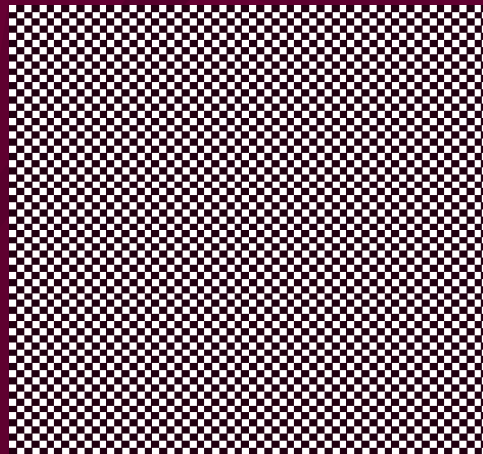
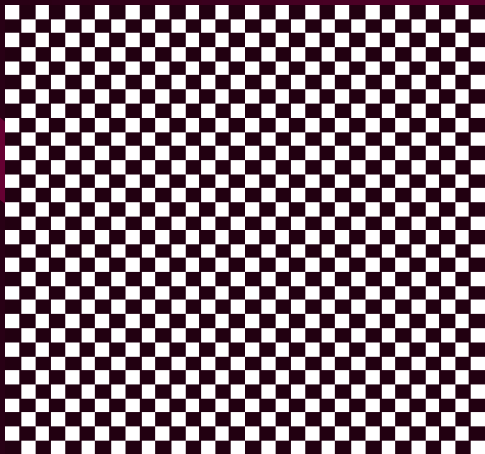
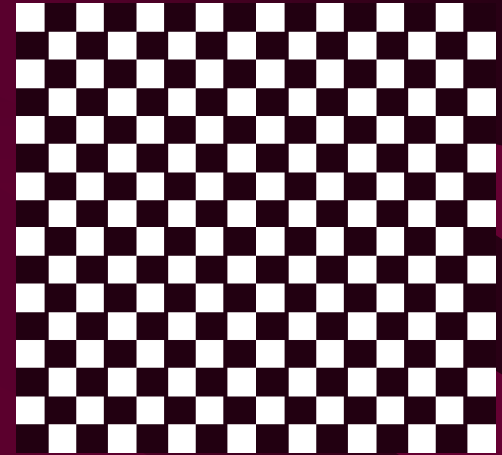
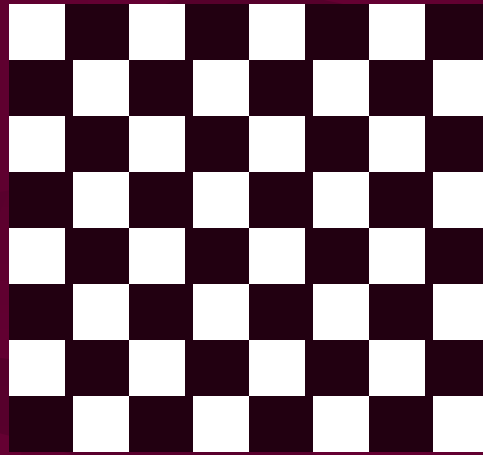
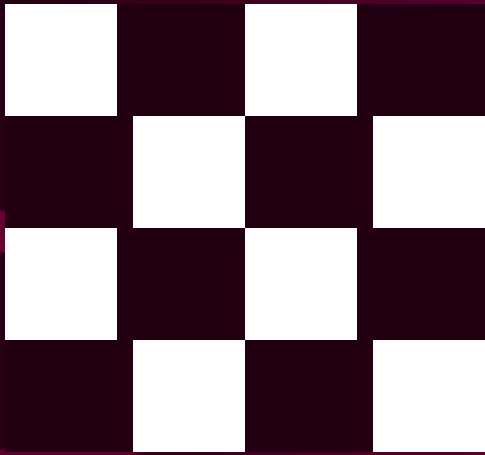




Dithering

- Why does it work? *Spatial integration*. Using the fact that our eyes usually blend areas of high frequency.
- Why do you need it? If you don't have enough bits. Eyes can detect 100 shades of a color. Banding occurs with fewer colors.
- Where do you see this? Printers and newspaper! Let's look at the Alligator.

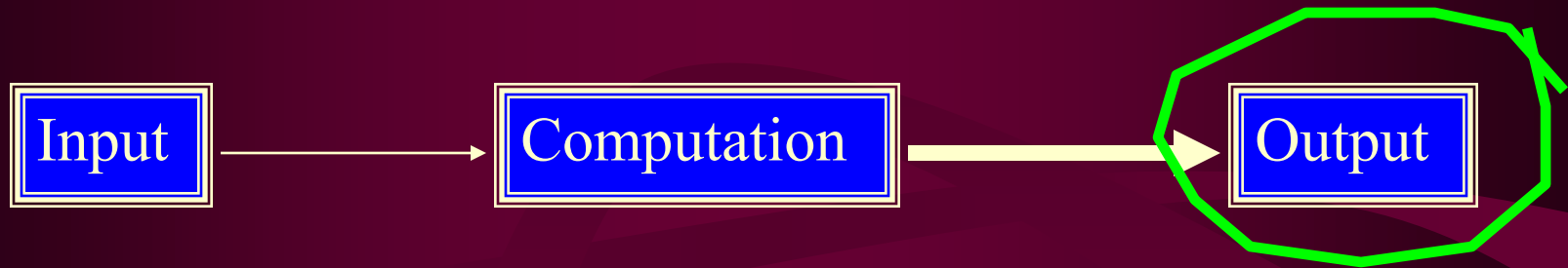
Black and White Dithering



Let's transition to OpenGL

- Now that we understand the input and computation stage, let's hold off on different output types till after `opengl + 2D`

Output



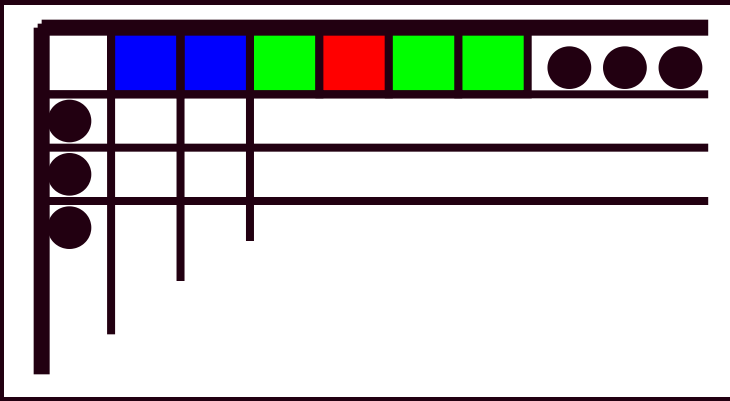
We have an image (framebuffer or model), now we want to show it. Read Ch. 2 in the Hearn and Baker handout.

- Hardcopy
- Display
 - Vector
 - Raster Scan

Hardcopy

- Printers (Resolution, color depth)
 - Dot Matrix - uses a head with 7 to 24 pins to strike a ribbon (single or multiple color)
 - Ink Jet Printers (fires small balls of colored ink)
 - Laser Printers (powder adheres to positive charged paper)
 - Pen Plotters (similar to vector displays).
“infinite” resolution.

Framebuffer -> Monitor



The values in the framebuffer are converted from a digital (1s and 0s representation, the bits) to an analog signal that goes out to the monitor. A video card's RAMDAC performs this operation, once per frame. This is done automatically (not controlled by your code), and the conversion can be done while writing to the framebuffer.

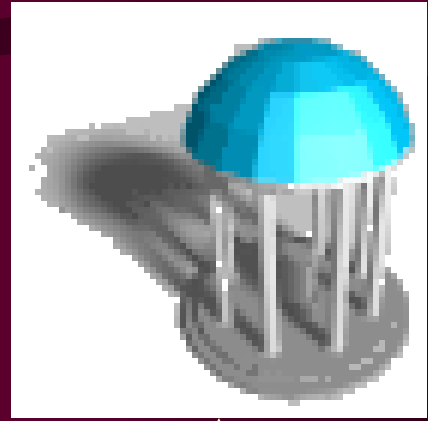
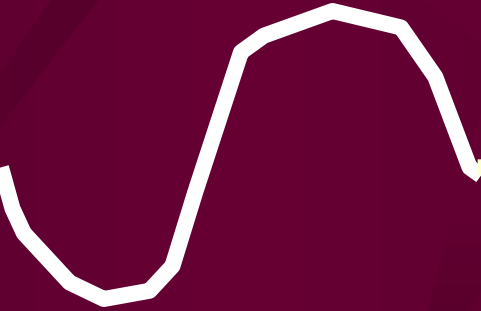
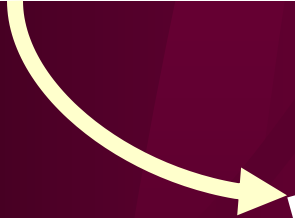
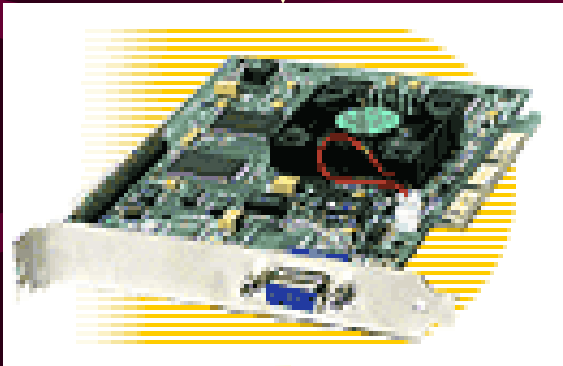
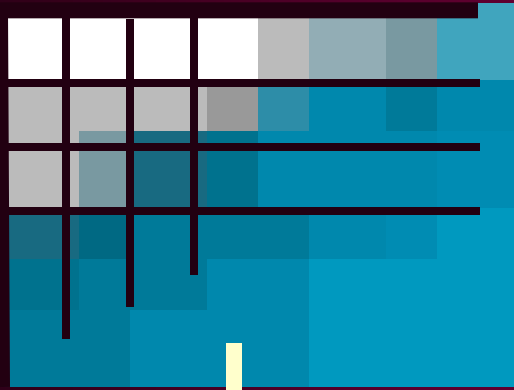


Image Quality Issues

- Screen resolution
- Color
- Blank space between the pixels
- Intentional image degradation
- Brightness
- Contrast
- Refresh rate
- Sensitivity of display to viewing angle

Pixels

- Pixel - The most basic addressable image element in a screen
 - CRT - Color triad (RGB phosphor dots)
 - LCD - Single color element
- Screen Resolution - measure of number of pixels on a screen (m by n)
 - m - Horizontal screen resolution
 - n - Vertical screen resolution

Other meanings of resolution

- Pitch - Size of a pixel, distance from center to center of individual pixels.
- Cycles per degree - Addressable elements (pixels) divided by twice the FOV measured in degrees.
- The human eye can resolve 30 cycles per degree (20/20 Snellen acuity).

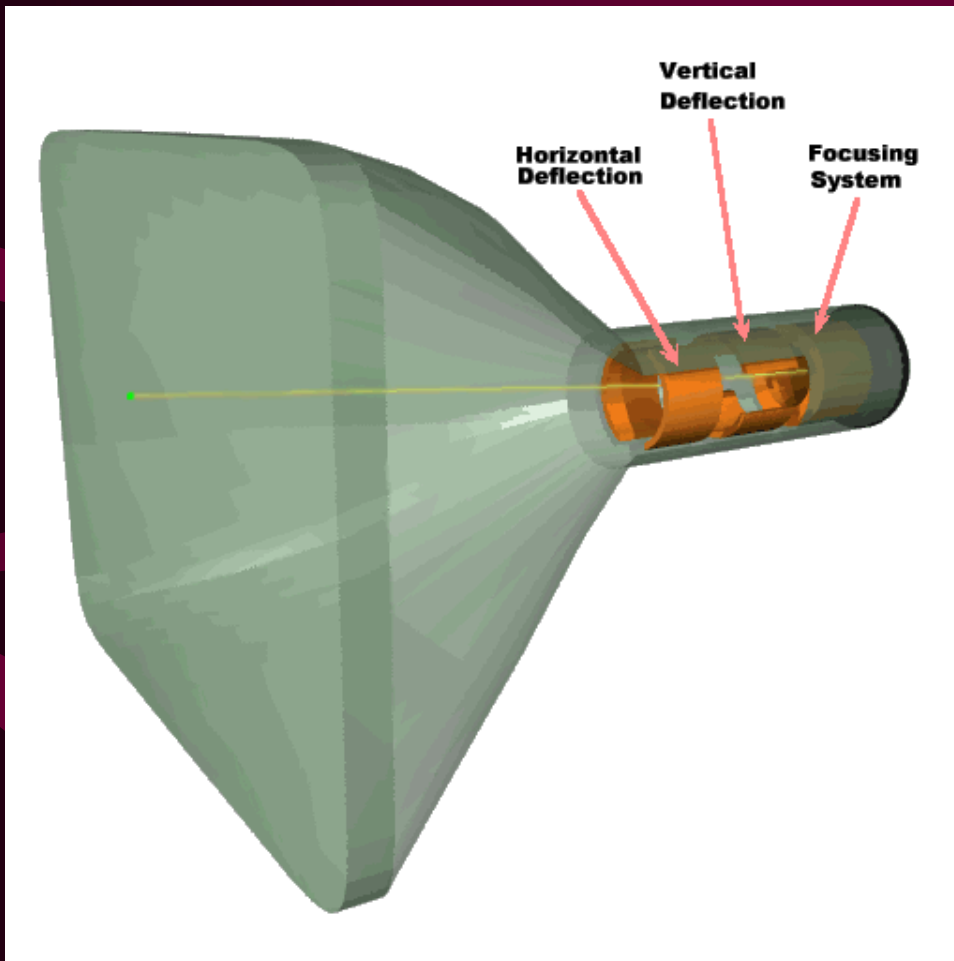
Video Formats

- **NTSC** - 525x480, 30f/s, interlaced
- **PAL** - 625x480, 25f/s, interlaced
- **VGA** - 640x480, 60f/s, noninterlaced
- **SVGA** – 800x600, 60f/s noninterlaced
- **RGB** - 3 independent video signals and synchronization signal, vary in resolution and refresh rate
- **Time-multiplexed color** - R,G,B one after another on a single signal, vary in resolution and refresh rate

Raster Displays

- Cathode Ray Tubes (CRTs), most “tube” monitors you see. Very common, but big and bulky.
- Liquid Crystal Displays (LCDs), there are two types transmissive (laptops, those snazzy new flat panel monitors) and reflective (wrist watches).

Cathode Ray Tubes (CRTs)



Heating element on the *yolk*.

Phosphor coated screen

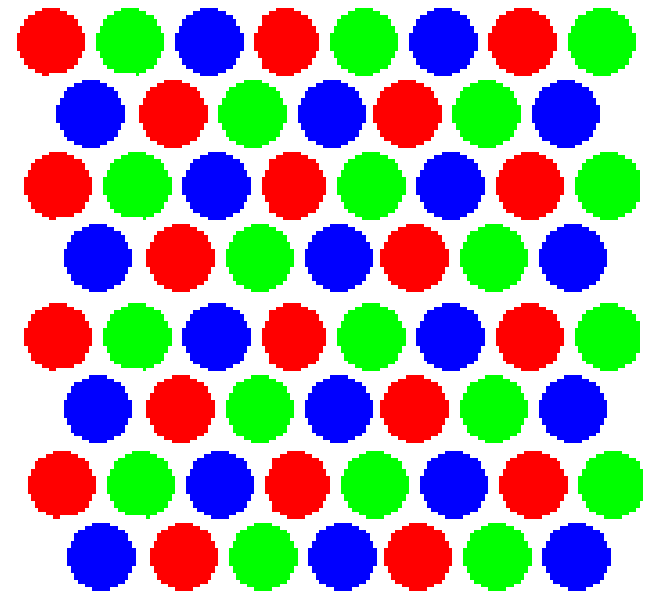
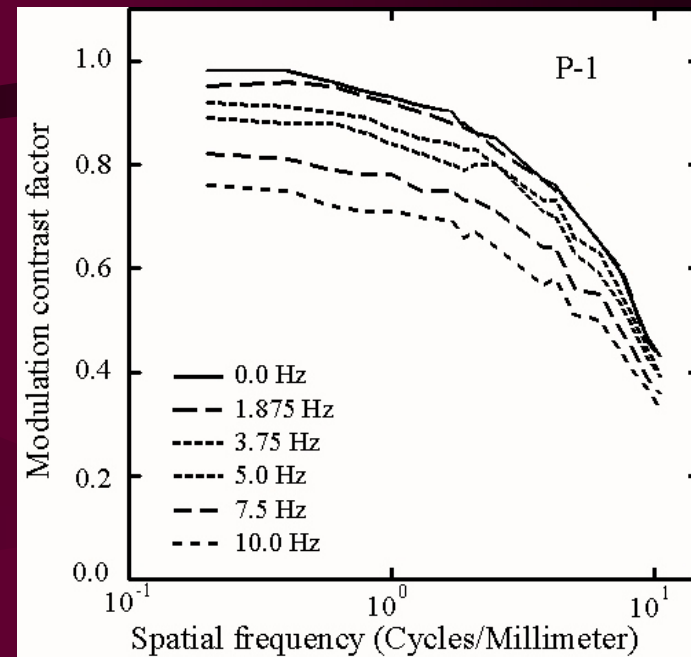
Electrons are boiled off the filament and drawn to the focusing system.

The electrons are focused into a beam and “shot” down the cylinder.

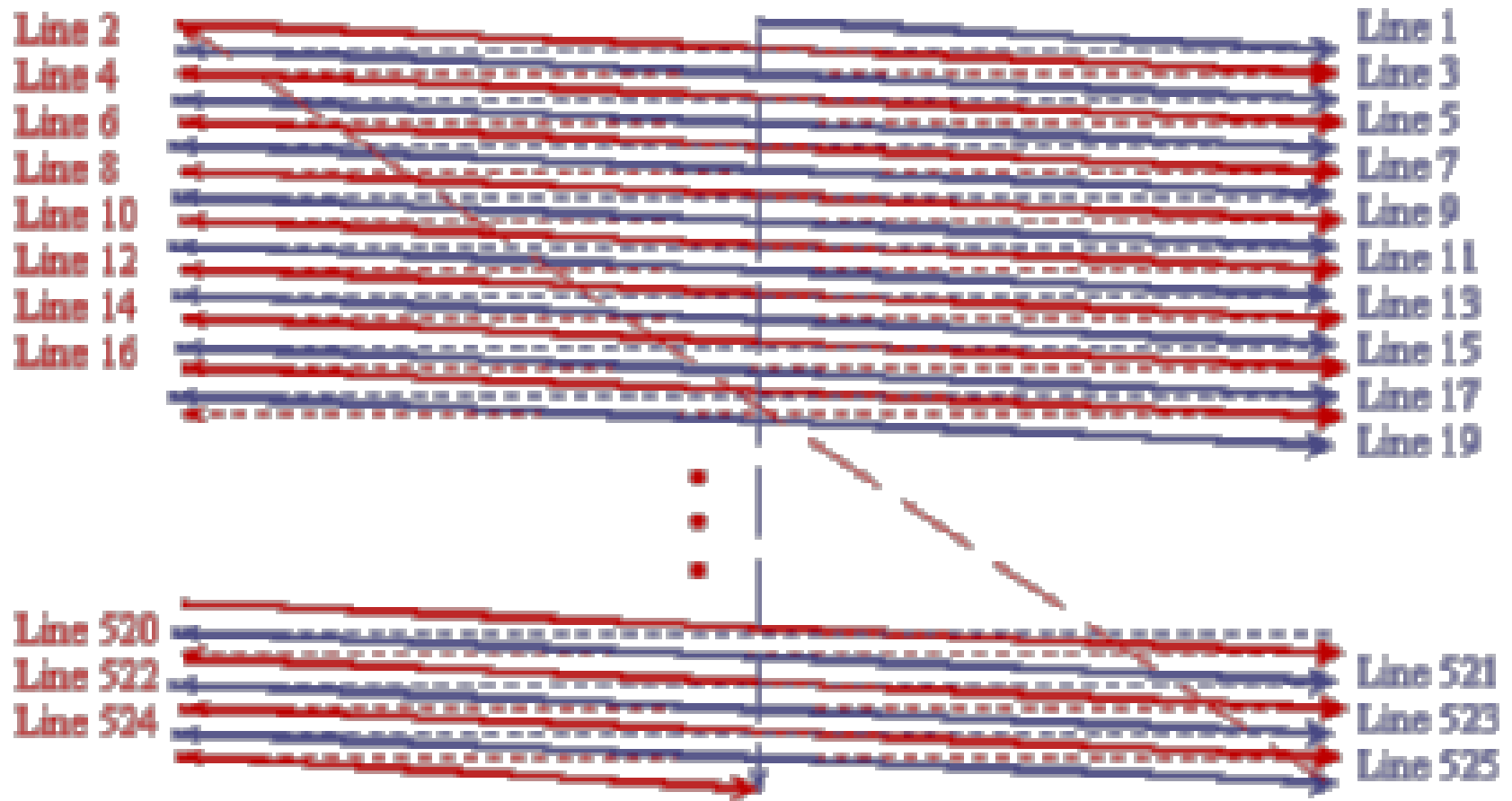
The deflection plates “aim” the electrons to a specific position on the screen.

CRT Phosphor Screen

- The screen is coated with phosphor, 3 colors for a color monitor, 1 for monochrome.
- For a color monitor, three guns light up red, green, or blue phosphors.
- Intensity is controlled by the amount of time at a specific phosphor location.



Beam Movement



Beam Movement

scan line - one row on the screen

interlace vs. non-interlace - Each frame is either drawn entirely, or as two consecutively drawn fields that alternate horizontal scan lines.

vertical sync (vertical retrace) - the motion of the beam moving from the bottom of the image to the top, after it has drawn a frame.

refresh rate - how many frames are drawn per second. Eye can see 24 frames per second. TV is 30 Hz, monitors are at least 60 Hz.

- *Refresh rate* is important, but remember it is different than your program's *update rate*.
- Why is higher, better?

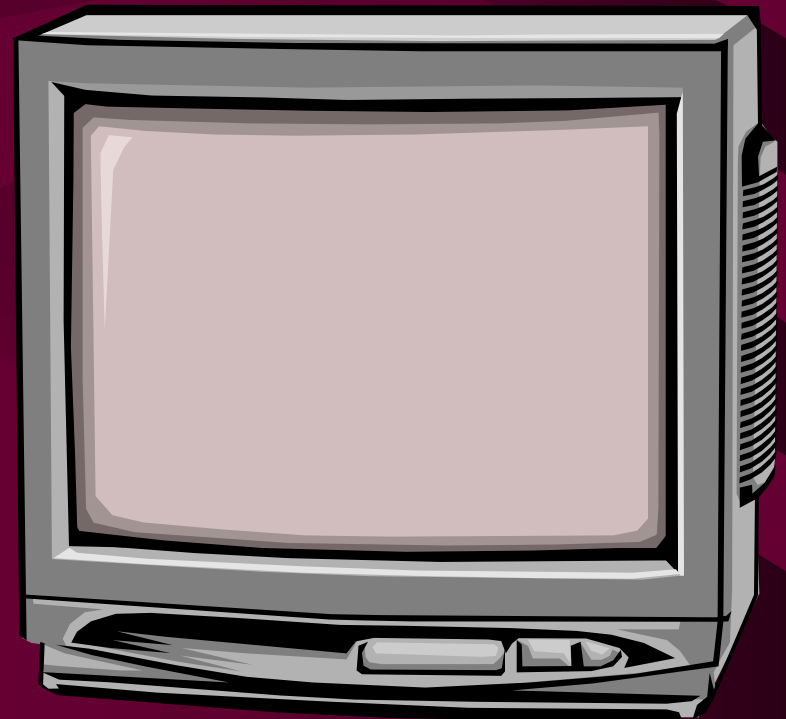
Vector Displays

- Unlike CRTs, vector displays have a single gun that is controlled to draw lines. Think of having a VERY FAST drawing pen.
- Pros: Diagrams/only draw what you need
- Cons: No fill objects/Slows with complexity



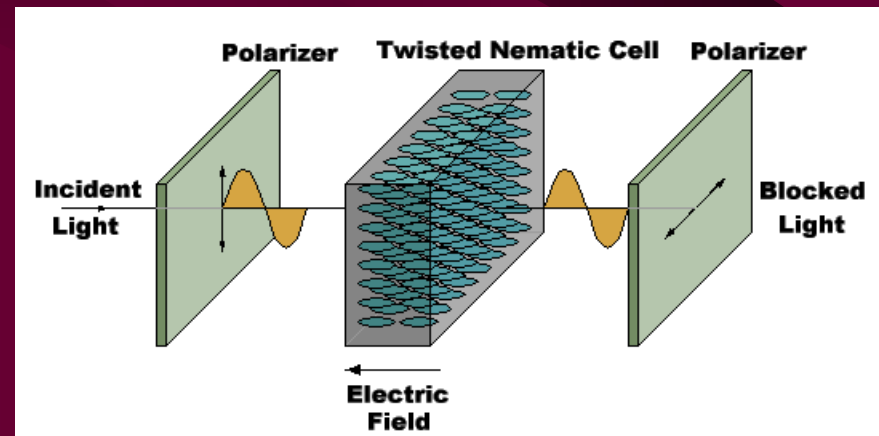
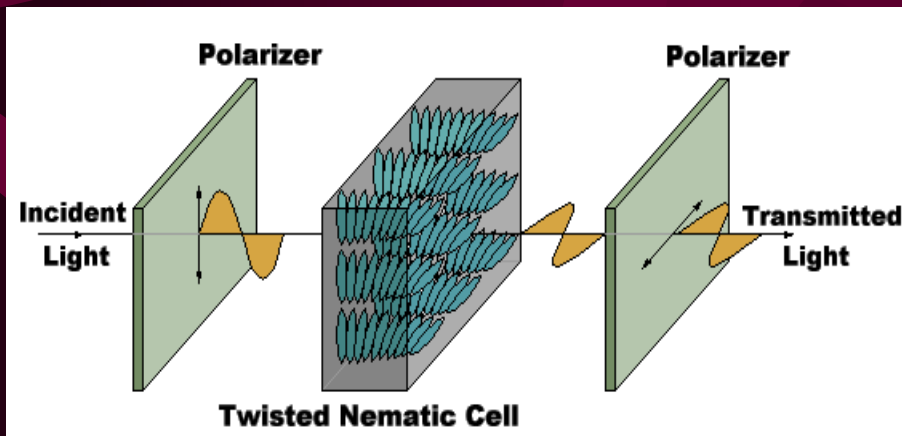
CRTs (cont.)

- Strong electrical fields and high voltage
- Very good resolution
- Heavy, not flat



Liquid Crystal Displays (LCDs)

- Also divided into pixels, but without an electron gun firing at a screen, LCDs have cells that either allow light to flow through, or block it.



Liquid Crystal Displays

- Liquid crystal displays use small flat chips which change their transparency properties when a voltage is applied.
- LCD elements are arranged in an $n \times m$ array call the LCD matrix
- Level of voltage controls gray levels.
- LCDs elements do not emit light, use backlights behind the LCD matrix

LCDs (cont.)

- Color is obtained by placing filters in front of each LCD element
- Usually black space between pixels to separate the filters.
- Because of the physical nature of the LCD matrix, it is difficult to make the individual LCD pixels very small.
- Image quality dependent on viewing angle.

Advantages of LCDs

- Flat
- Lightweight
- Low power consumption



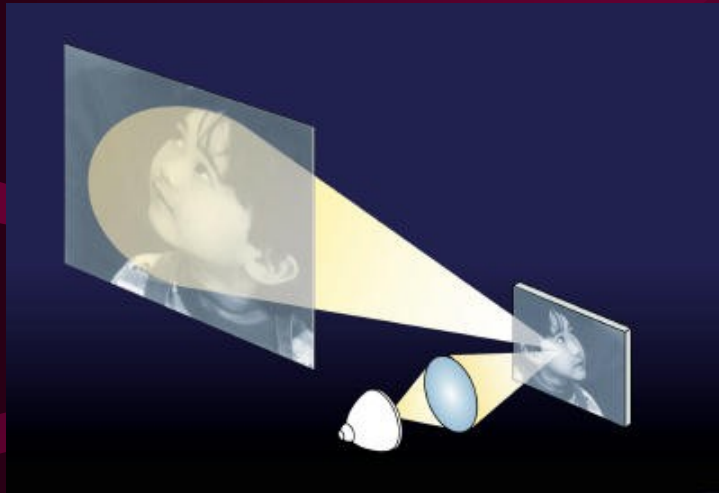
Projection Displays

- Use bright CRT or LCD screens to generate an image which is sent through an optical system to focus on a (usually) large screen.

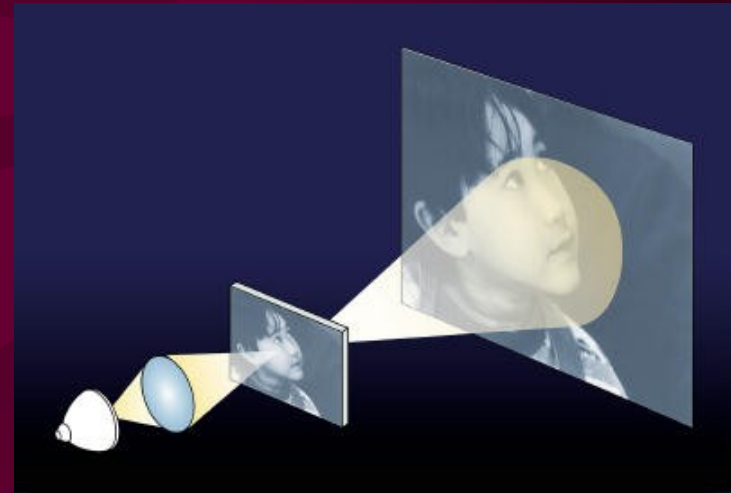


Basic Projector Designs

(Images from Phillips Research)



Reflective Projection System



Transmissive Projection System

Transmissive Projectors



CRT-based projectors are usually heavy and large compared to other technologies

CRT Based

- One color CRT tube (red, blue, green phosphors) displays an image with one projection lens.
- One black-and-white CRT with a rapidly rotating color filter wheel (red, green, blue filters) is placed between the CRT tube and the projection lens.
- Three CRT tubes (red, green, blue) with three lenses project the images. The lenses are aligned so that a single color image appears on the screen.

Transmitive Projectors

- LCD Based
 - Use a bright light to illuminate an LCD panel, and a lens projects the image formed by the LCD onto a screen.
- Small, lightweight compared to CRT based displays

Reflective Projectors

- In reflective projectors, the image is formed on a small, reflective chip.
- When light shines on the chip, the image is reflected off it and through a projection lens to the screen.
- Recent innovations in reflective technology have been in the the following areas:
 - **Microelectromechanical systems (MEMS)**
 - Digital micromirror device (DMD, DLP)
 - Grating light valve (GLV)
 - **Liquid crystal on silicon (LCOS)**

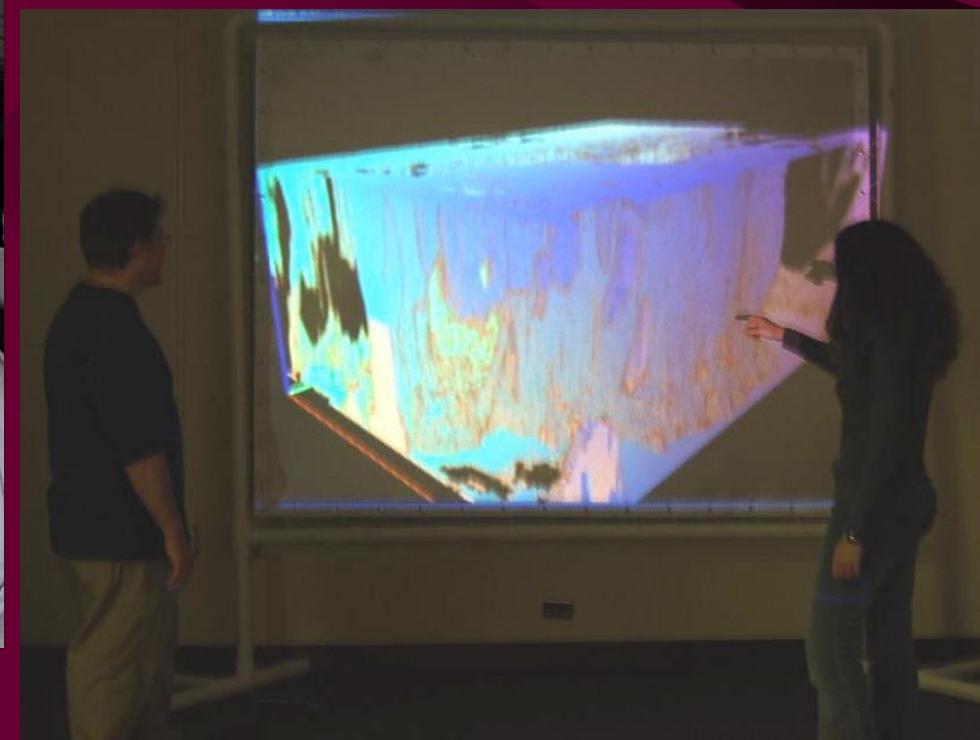
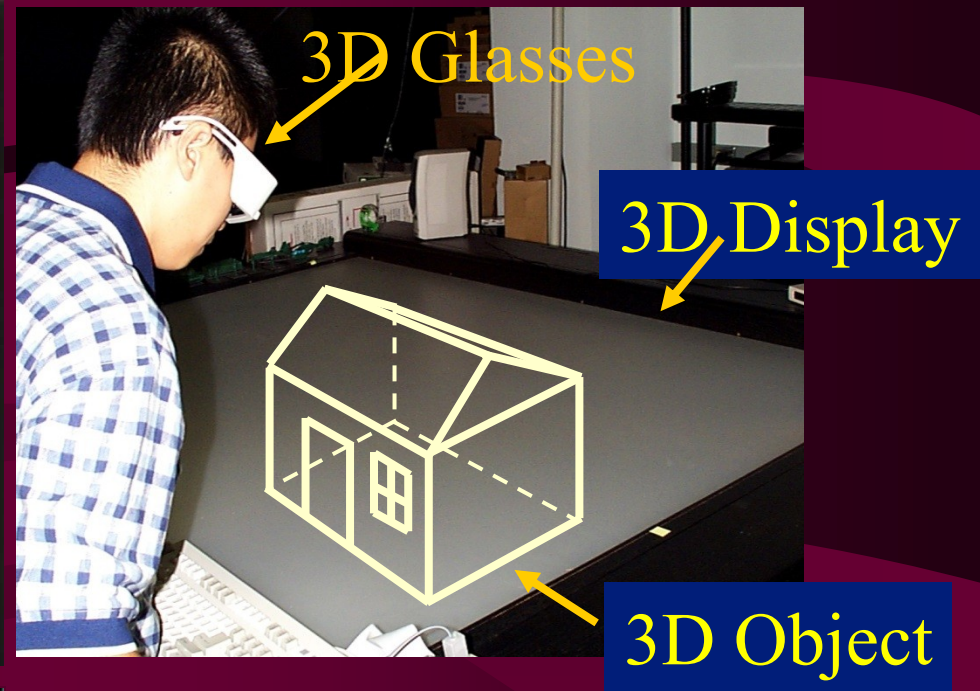
Advantages/Disadvantages of Projection Display

- Very large screens can provide large FoV and can be seen by several people simultaneously.
- Image quality can be fuzzy and somewhat dimmer than conventional displays.
- Sensitivity to ambient light.
- Delicate optical alignment.

Displays in Virtual Reality

- Head-Mounted Displays (HMDs)
 - The display and a position tracker are attached to the user's head
- Head-Tracked Displays (HTDs)
 - Display is stationary, tracker tracks the user's head relative to the display.
 - Example: CAVE, Workbench, Stereo monitor





Graphics Software

- How to talk to the hardware?
- Special purpose software
 - Excel
 - AutoCAD
 - Medical Visualization
- Programming API
- Advantages?
- Please Read Section 2.9