Moore’s Law

“Chip density doubles every 18 months.”

Processing Power ($P$) in 15 years:

\[
P = P_{today} (2)^{\frac{15 \text{ years}}{18 \text{ months}}} = P_t (2)^{\frac{15}{1.5}}
\]

\[
= P_t (2)^{10} = 1000P_t
\]
Ivan Sutherland - 1963
Cornell in Perspective Film
Direct Illumination
• **Environment**
  - Geometry & topology
  - Material properties
    - Color, reflectance, textures
    - (Cost, strength, thermal properties)

• **Lighting**
  - Geometry & position
  - Intensity, spectral distribution
  - Direction, spatial distribution
Camera

- Viewer Position
- Viewer direction
- Field of view
  - Wide angle
  - Telephoto
- Depth of focus
  - Near
  - Far
Perspective Transformation

- Perspective transformation
  Matrix multiplication

- Clipping

- Culling
The concept of the picture plane may be better understood by looking through a window or other transparent plane from a fixed viewpoint. Your lines of sight, the multitude of straight lines leading from your eye to the subject, will all intersect this plane. Therefore, if you were to reach out with a grease pencil and draw the image of the subject on this plane you would be “tracing out” the infinite number of points of intersection of sight rays and plane. The result would be that you would have “transferred” a real three-dimensional object to a two-dimensional plane.
Hidden Line Algorithm
Hidden Line Algorithm
Mapping a Viewing Frustum to a Standard Viewbox

Frustum of vision

Screen coordinate system

© Donald P. Greenberg - Cornell Program of Computer Graphics
Perspective Projections in Computer Graphics

Eye coordinate system

Plan or elevation view
Object Space

(One point perspective)
Image Space

(One point perspective)
Raster Operations

- Conversion from polygons to pixels
- Hidden surface removal (z-buffer)
- Incremental shading
Visible Surface Algorithm
Z-Buffer Algorithm

1. Set $\text{depth}(x, y) = 1.0$
   $\text{intensity}(x,y) = \text{background color}$

2. For each polygon, find all pixels covered

3. Calculate $z(x,y)$ of each pixel covered by the polygon

4. If $z(x,y) < \text{depth}(x,y)$, polygon is closer
   set $\text{depth}(x,y) = z(x,y)$
   change color
Depth Buffer Algorithm
Depth Buffer Algorithm
### Phong Reflection Model

**SPHERES - phong model - gamma corrected
Ka=.2**

<table>
<thead>
<tr>
<th>$K_s/K_d$</th>
<th>100/0</th>
<th>75/25</th>
<th>50/50</th>
<th>25/75</th>
<th>0/100</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N_s=5$</td>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
<td><img src="image5.png" alt="Image" /></td>
</tr>
<tr>
<td>$N_s=10$</td>
<td><img src="image6.png" alt="Image" /></td>
<td><img src="image7.png" alt="Image" /></td>
<td><img src="image8.png" alt="Image" /></td>
<td><img src="image9.png" alt="Image" /></td>
<td><img src="image10.png" alt="Image" /></td>
</tr>
<tr>
<td>$N_s=20$</td>
<td><img src="image11.png" alt="Image" /></td>
<td><img src="image12.png" alt="Image" /></td>
<td><img src="image13.png" alt="Image" /></td>
<td><img src="image14.png" alt="Image" /></td>
<td><img src="image15.png" alt="Image" /></td>
</tr>
<tr>
<td>$N_s=40$</td>
<td><img src="image16.png" alt="Image" /></td>
<td><img src="image17.png" alt="Image" /></td>
<td><img src="image18.png" alt="Image" /></td>
<td><img src="image19.png" alt="Image" /></td>
<td><img src="image20.png" alt="Image" /></td>
</tr>
</tbody>
</table>
Image Storage

- Typical frame buffer
  - 1280 x 1024 pixels
  - 3 channels (red, green, blue)
  - 1 byte/channel
- Total memory
  - 3 3/4 megabytes - single buffer
  - 7 1/2 megabytes - double buffer
• Digital to analog conversion
  1920 x 1080 resolution
  60 frames per second

• Total data rate
  2 million pixels
  x 3 bytes/pixel
  x 60 frames/second
  = 360 megabytes/second
Refresh vs. Update Rate

- The “refresh rate” is the number of times per second the entire image is drawn.

- The “update rate” is the number of times per second the image is changed.
Direct Illumination
Why a Pipeline?

A pipeline allows multiple processes to occur in parallel. Example:

• An automobile assembly line. Assume 4 stations, each taking 2 minutes to do its task. It takes 8 minutes to make a car, but the rate at which cars are made is one every 2 minutes.
Example: Automobile Pipeline

Automobile takes 8 minutes to make, but the assembly line makes a car every two minutes.

- Build frame: 2 minutes
- Add engine: 2 minutes
- Add windows: 2 minutes
- Paint body: 2 minutes
Automobile takes 8 minutes to make, but the assembly line makes a car every two minutes.
E & S Frame Buffer 1975
Texture Mapping Units - TMU’s

- A system with one TMU extracts the appropriate texel or texels from texture memory, minifies or magnifies it and filters it
- The texturecombine unit can scale the result
• Hardware architectures now support accessing more than one texture in a single pass
Multipass Example: Light Maps

• Two separate textures, one for the material’s composition, one for the lighting

J.L. Mitchell, M. Tatro, and I. Bullard
Castle’s Geometry

Agata & Andrzej Wojaczek, Advanced Graphics Applications Inc.
Reflection Example - Castle
Putting it all together

Gloss textures on pear, shadows on curved surfaces, reflections dropping off with depth from table.

J.L. Mitchell & E. Hart, ATI Technologies, Inc.
Graphics Pipeline - 1980's

M — Model
L — Lighting
P — Perspective/Clipping
S — Scan Conversion/Z-buffer
D — Display Storage
V — Video
Graphics Pipeline - 2000 +

M — Model
L — Lighting
P — Perspective/Clipping
T — Texturing
S — Scan Conversion/Z-buffer
D — Display Storage
V — Video
Graphics Hardware Recap

1970

Model → P-xform → Clipping → Lighting → Rasterize → System Memory → Film Recorder

1980

Model → P-xform → Clipping → Lighting → Rasterize → Frame Buffer → Display

1986

Model → Lighting → P-xform → Clipping → Rasterize → Z-Buffer → Frame Buffer → Display

1999

Model → Lighting → P-xform → Clipping → Texture Mapping → Rasterize → Z-Buffer → Frame Buffer → Display
Graphics Hardware Recap

2000

CPU

Model

Vertex Buffer → Lighting → P-xform Clipping → Texture Mapping

Z-Buffer

Frame Buffer

Graphics Hardware

2001

CPU

Model

Vertex Buffer → Lighting → P-xform Clipping → Texture Mapping

Vertex Operations

Nvidia

Rasterize

Z-Buffer

Frame Buffer

Display

Pixel Operations
Interesting Trends

• Moore’s Law is approximately 1.7x increase in speed a year; graphics accelerators are improving at 2x to 4x a year

• Reducing energy (watts) is more important than increasing processing speed (flops)
“Moore's Law is for wimps.”
Faster than Moore’s Law

One-pixel polygons (~10M polygons @ 30Hz)

Slope ~2.4x/year
( Moore’s Law ~ 1.7x/year)

Graph courtesy of Professor John Poulton (from Eric Haines)
GeForce Transistor Count and Semiconductor Process

<table>
<thead>
<tr>
<th>GPU</th>
<th>Transistor Count (Millions)</th>
<th>Process (µm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Riva ZX</td>
<td></td>
<td>0.35</td>
</tr>
<tr>
<td>Riva TNT2</td>
<td></td>
<td>0.22</td>
</tr>
<tr>
<td>GeForce2 GTS</td>
<td></td>
<td>0.18</td>
</tr>
<tr>
<td>GeForce3</td>
<td></td>
<td>0.18</td>
</tr>
<tr>
<td>GeForce4 Ti 4600</td>
<td></td>
<td>0.15</td>
</tr>
<tr>
<td>GeForce FX</td>
<td></td>
<td>0.13</td>
</tr>
<tr>
<td>GeForce 6800 Ultra</td>
<td></td>
<td>0.13</td>
</tr>
<tr>
<td>GeForce 7800 GTX</td>
<td></td>
<td>0.11</td>
</tr>
</tbody>
</table>
NVIDIA’s GK110 Kepler Chip 2012

- 7.1 billion transistors
- 2880 processor cores
- 28 nanometers
- PCI Express Gen3
NVIDIA’s new Maxwell Chip 2014

- 6144 processor cores
- 20 nm
<table>
<thead>
<tr>
<th></th>
<th>“SANDY BRIDGE”</th>
<th>“IVY BRIDGE”</th>
<th>“HASWELL” Estimated</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPU</td>
<td>17%</td>
<td>27%</td>
<td>31%</td>
</tr>
</tbody>
</table>

*Estimated values.*
ELITE AMD A-SERIES / CODENAMED "RICHLAND"

42% GPU*
AMD – Integrated Graphics 2014

- “Kaveri”
- 28 nm
- 47% GPU
Processing Power Compared

- 2015: iPhone 5 > 1985 Cray-2 (2.7x)
Nvidia’s Maxwell Architecture

**MAXWELL “GM204” Top Level**

- 5.2 Billion Transistors
- 2x performance vs GK104
- 16 SMM
- 2048 CUDA Cores
- 16 Geometry Units
- 128 Texture Units
- 64 ROP Units
- 256-bit GDDR5
<table>
<thead>
<tr>
<th>Processor</th>
<th>Transistor count</th>
<th>Date of introduction</th>
<th>Manufacturer</th>
<th>Process</th>
<th>Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>R520</td>
<td>321,000,000</td>
<td>2005</td>
<td>AMD</td>
<td>90 nm</td>
<td>288 mm²</td>
</tr>
<tr>
<td>R580</td>
<td>384,000,000</td>
<td>2006</td>
<td>AMD</td>
<td>90 nm</td>
<td>352 mm²</td>
</tr>
<tr>
<td>G80</td>
<td>681,000,000</td>
<td>2006</td>
<td>NVIDIA</td>
<td>90 nm</td>
<td>480 mm²</td>
</tr>
<tr>
<td>R600 Pele</td>
<td>700,000,000</td>
<td>2007</td>
<td>AMD</td>
<td>80 nm</td>
<td>420 mm²</td>
</tr>
<tr>
<td>G92</td>
<td>754,000,000</td>
<td>2007</td>
<td>NVIDIA</td>
<td>65 nm</td>
<td>324 mm²</td>
</tr>
<tr>
<td>RV790XT Spartan</td>
<td>959,000,000</td>
<td>2008</td>
<td>AMD</td>
<td>55 nm</td>
<td>282 mm²</td>
</tr>
<tr>
<td>GT200 Tesla</td>
<td>1,400,000,000</td>
<td>2008</td>
<td>NVIDIA</td>
<td>65 nm</td>
<td>576 mm²</td>
</tr>
<tr>
<td>Cypress RV870</td>
<td>2,154,000,000</td>
<td>2009</td>
<td>AMD</td>
<td>40 nm</td>
<td>334 mm²</td>
</tr>
<tr>
<td>Cayman RV970</td>
<td>2,640,000,000</td>
<td>2010</td>
<td>AMD</td>
<td>40 nm</td>
<td>389 mm²</td>
</tr>
<tr>
<td>GF100 Fermi</td>
<td>3,200,000,000</td>
<td>Mar 2010</td>
<td>NVIDIA</td>
<td>40 nm</td>
<td>526 mm²</td>
</tr>
<tr>
<td>GF110 Fermi</td>
<td>3,000,000,000</td>
<td>Nov 2010</td>
<td>NVIDIA</td>
<td>40 nm</td>
<td>520 mm²</td>
</tr>
<tr>
<td>GK104 Kepler</td>
<td>3,540,000,000</td>
<td>2012</td>
<td>NVIDIA</td>
<td>28 nm</td>
<td>294 mm²</td>
</tr>
<tr>
<td>Tahiti RV1070</td>
<td>4,312,711,873</td>
<td>2011</td>
<td>AMD</td>
<td>28 nm</td>
<td>365 mm²</td>
</tr>
<tr>
<td>GK110 Kepler</td>
<td>7,080,000,000</td>
<td>2012</td>
<td>NVIDIA</td>
<td>28 nm</td>
<td>561 mm²</td>
</tr>
<tr>
<td>RV1090 Hawaii</td>
<td>6,300,000,000</td>
<td>2013</td>
<td>AMD</td>
<td>28 nm</td>
<td>438 mm²</td>
</tr>
<tr>
<td>GM204 Maxwell</td>
<td>5,200,000,000</td>
<td>2014</td>
<td>NVIDIA</td>
<td>28 nm</td>
<td>398 mm²</td>
</tr>
<tr>
<td>GM200 Maxwell</td>
<td>8,100,000,000</td>
<td>2015</td>
<td>NVIDIA</td>
<td>28 nm</td>
<td>601 mm²</td>
</tr>
<tr>
<td>Fiji</td>
<td>8,900,000,000</td>
<td>2015</td>
<td>AMD</td>
<td>28 nm</td>
<td>596 mm²</td>
</tr>
</tbody>
</table>

Moore’s Law – GPU Transistor Counts

Moore's Law

– GPU Transistor Counts

http://en.wikipedia.org/wiki/Transistor_count
End...