History of Computer GRAPHICS/GAMES

Sandro Spina
Computer Science Department
2012

Components of CG

- **Hardware** (CPU, GPU, etc.)
- **Software** (2D, 3D, Z-Buffering, Rendering Algorithms, etc.)
- **Applications** (Games, Films, Simulations, etc.)

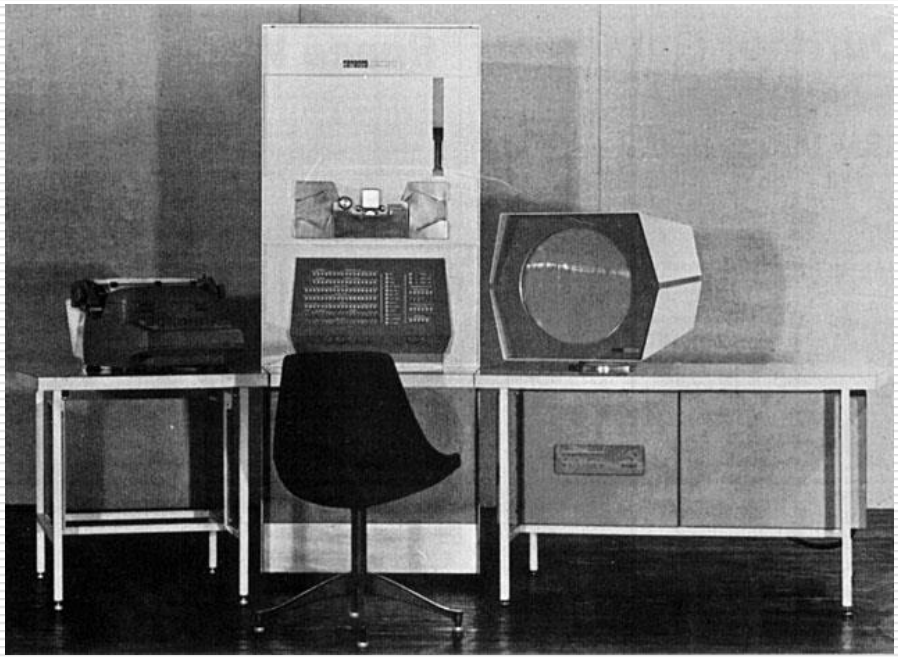
Vector / Raster Graphics

- Vector graphics are composed of thin lines drawn between coordinates. Most 70s and mid-80s computers used vector graphics. Remember Asteroids Game.

- Nowadays video displays transform vector representations into a raster format image, which contains a value for every pixel on the screen. Image is re-drawn according to the monitor refresh rate.

The PDP-1 (1960)

- Programmed Data Processor.
- The World's First Toy **Computer**
- The game *Spacewar* was developed on this computer for demonstration purposes.



First Computer Game

- 1961 – Steve Russell at MIT created the first computer game, **Spacewar**

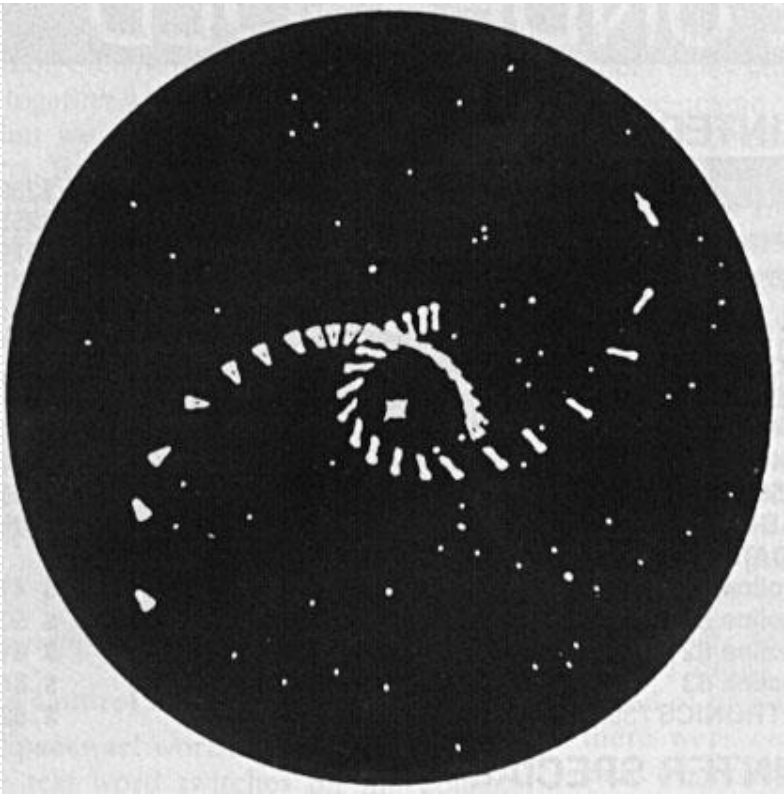
- **Basic Rules** - At least two spaceships, each controlled by a set of console switches. The ships have a supply of rocket fuel and weapons.

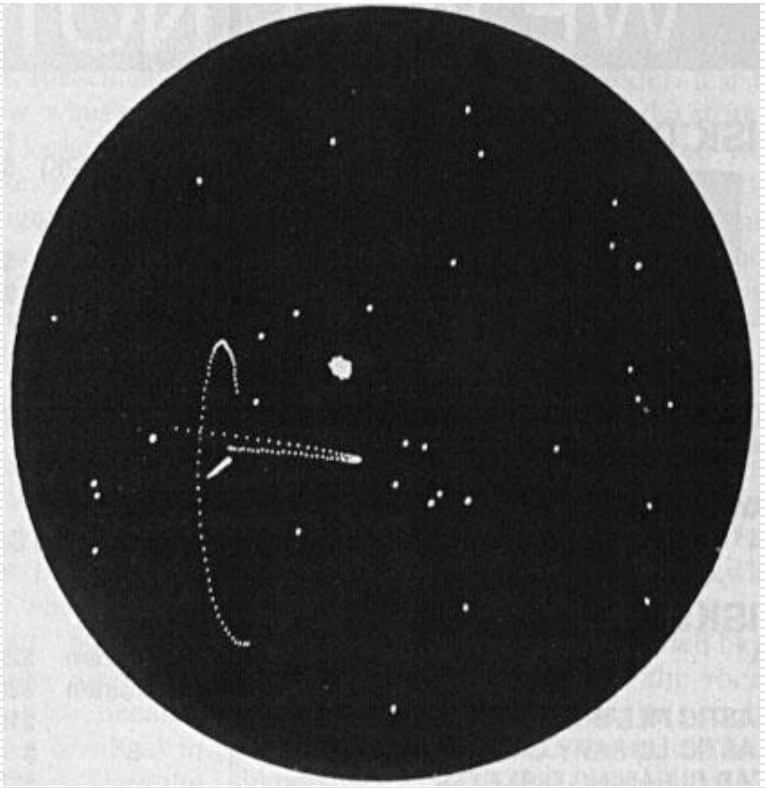
Spacewar – Game Strategy !!

- Initially the game used random stars as its background. However, this was immediately corrected (by Peter Samson) by encoding the familiar constellations into the game. This added more 'elegance' – more 'authentic' feel.

- **Strategy**: Added to the game with the introduction of *gravity* in the gameplay.

Screen Shots





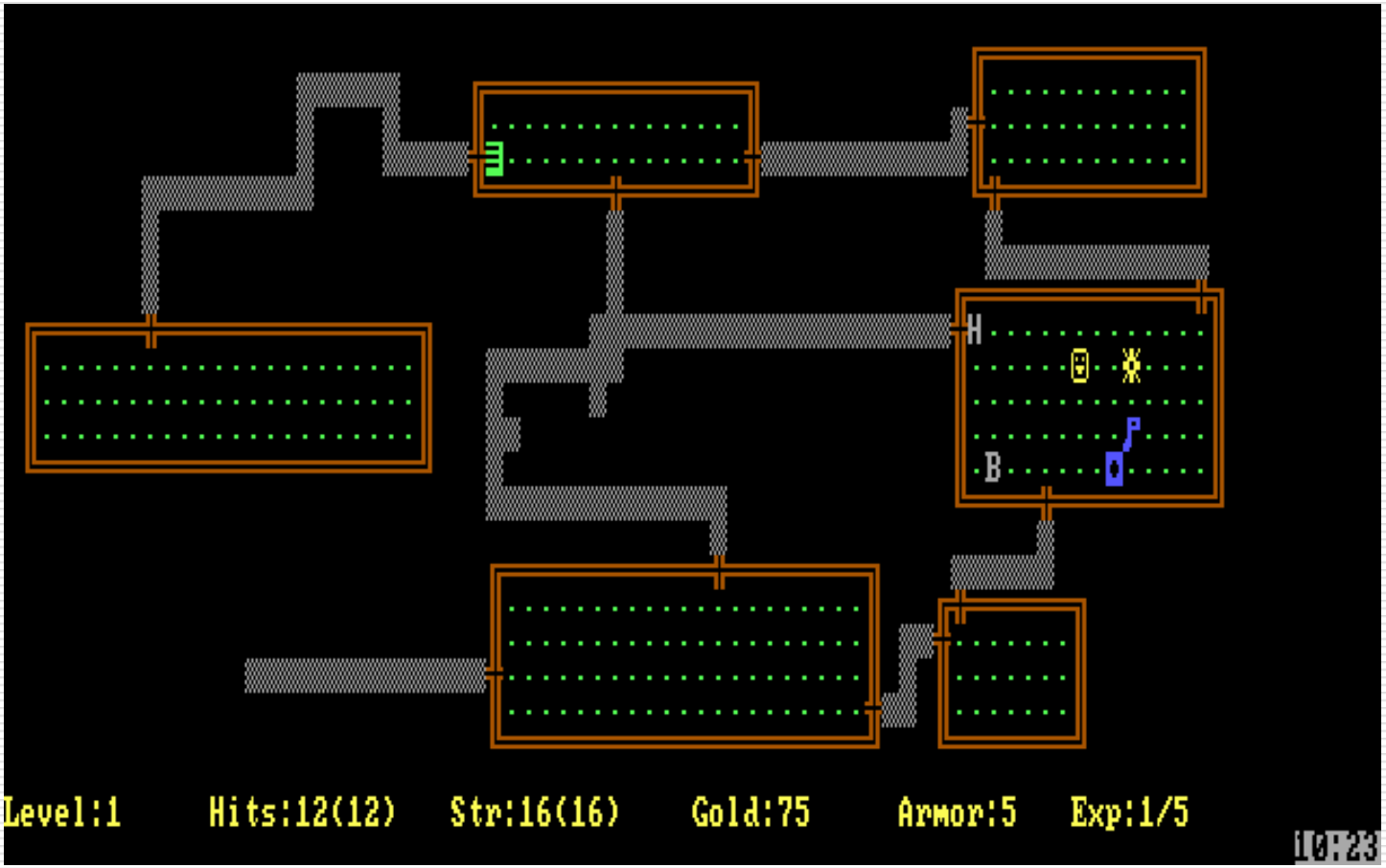
Text Based Games - Adventure

- Colossal Cave (ADVENT) was the first computer game in the genre of interactive fiction. Programmed in Fortran originally for the PDP-10.

- Sample text from the game :
  - *A huge green fierce snake bars the way!*
  - *With what? Your bare hands?* (refers to killing a dragon, etc.)
  - *It's not hungry (it's merely pinin' for the fjords)*

- Rogue (1980) / AngBand (1990)

Rogue(1980)



AngBand (1990)

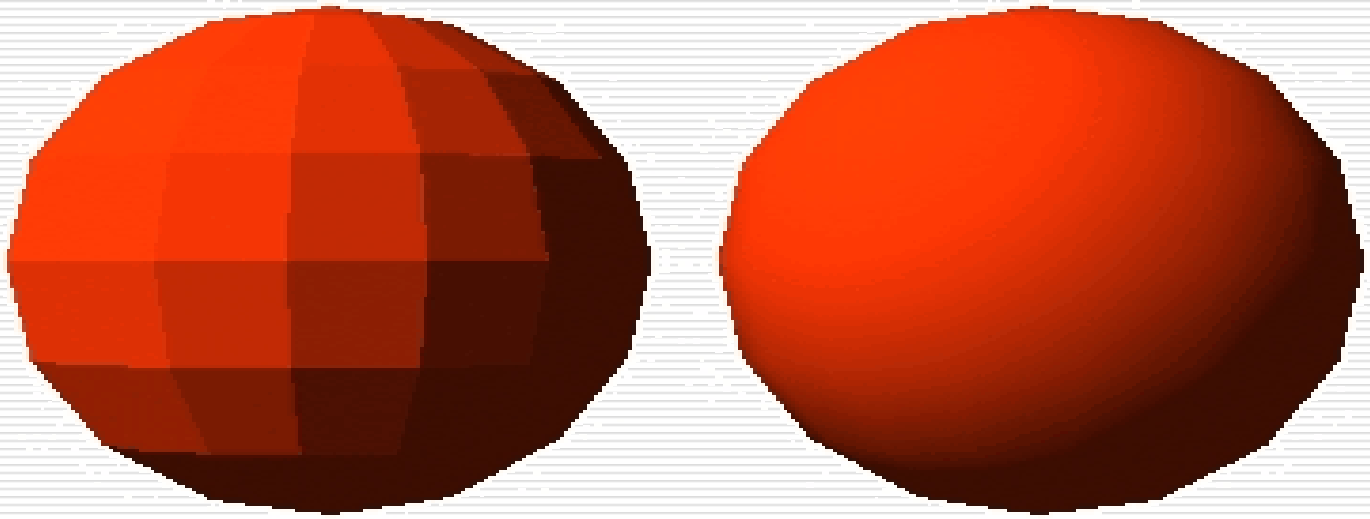
Graphics Rendering Pipeline

- The main function of the pipeline is to generate (or render), a two-dimensional image, given a virtual camera looking onto a three-dimensional scene consisting of three-dimensional objects.
- Optionally we might have light sources, shading equations, textures, participating media, etc.

- Rasterisation Based Techniques for Real-time Rendering
- Raytracing Based Techniques mostly for off-line Rendering
- Hybrids

1970s – Shading Algorithms

- Polygons meshes are used to represent objects.
- **Flat shading** makes an object look as if it's solid. Not very realistic !!
- This problem can be solved by creating objects with more smaller polygons. But this effects performance of the computer rendering the object. It also adds complexity to the objects.
- Gouraud Shading (1971) – presents a method for creating the appearance of a curved surface by interpolating the colour across the polygons. Colours are assigned to the vertices of the polygon. Rending is now more realistic !!



Phong Shading + Texture Mapping

- More complex version of shading, which was initially rarely used for real-time rendering, due to its 'computational complexity', is now used with most modern GPUs.
- Works by averaging the colour of each pixel by averaging the colours of the pixels adjacent to it.
- **Texture Mapping** applies 2D images to 3D objects. The algorithms work by 'stretching' the 2D image over the 3D object like a skin. Usually textures are applied to the polygon surfaces. Textures add realism to a computer generated scene.
- (Later 80s) The map image may not represent colour only but also describe transparency or other surface properties to be used in lighting or shading calculations.

Ray Tracing

- *Turner Whitted* (1980) came up with the idea of ray tracing. This was a new rendering method for simulating highly reflective surfaces. The algorithm works by tracing every ray of light, starting from the viewer's perspective back into the 3D scene to the objects. If an object happens to be reflective, the algorithm follows that ray of light as it bounces off the object until it hits something else (which is non-reflective)

- Ray tracing is clearly very computational intensive, however the realism that can be achieved is spectacular.

70's Major Events

- 1971 saw one of the most important advancements in GC – the microprocessor (CPU).
- In the 70's a number of animation houses were formed including Lucasfilm (George Lucas). Creators of the Star Wars films.
- 1973 was the setup of the Special Interest Group on Computer Graphics (ACM) and it's first major conference on computer graphics.
- In 1975 Mandelbrot describes Fractal Geometry. Fractals are used in computer graphics to create realistic simulations of natural phenomena such as mountains, coastlines, wood grain, etc.

GUIs

- January 1984, Apple Computers released the first Macintosh computer. It was the first personal computer to use a **graphical interface**.
- However, Xerox's Alto PC featured the first (WYSIWYG) editor with menus, icons, etc.
- IBM-compatible PCs
- Commodore launch the new AMIGA personal computer line. Based on Motorola's 68000 microprocessor.
- 1990 Microsoft ships Windows v3

1990s – Many more films

- Most notably is "Toy Story" (Disney and Pixar) which is the first commercially successful computer animated full length film.
- Terminator 2
- Beauty and the Beast (Many scenes contained 3D animated objects which were flat shaded with bright colours so as to blend with the hand-drawn characters)
- Batman
- Jurassic Park

QUAKE

- 1996 saw a breakthrough in the 3D gaming industry with the release of Quake (Carmack) by ID Software. 3D graphics accelerators started to pop up. Most notably the 3DFX “Voodoo” chipset.
- Used 3D models for players and monsters instead of 2 dimensional sprites and the world in which play takes place is created as a true 3-dimensional space.
- Unreal + Half Life

Open Graphics Library (OpenGL)

- It is a specification defining a cross-language cross-platform API for writing applications that produce 3D computer graphics (and 2D as well).
- Efficient implementation for OpenGL exist for Windows, many Unix platforms and the MacOS.
- Graphics cards (nowadays ATI and nVidia) support the OpenGL standard.

DirectX (Microsoft)

- Originally developed jointly between Microsoft and nVidia.
- It's main purpose was to allow all versions of Microsoft Windows, starting with Windows 95, to incorporate quality multimedia and computer gaming into the Windows environment.
- Nowadays we see the API battles between OpenGL and DirectX. Really what can be done with one API can be done with the other. Performance depends also on graphics card implementing the APIs.
- Microsoft is probably taking the lead in terms of usability.

Present Situation

- Film industry
  - Pixar's The Incredibles, Toy Story series, Up, etc.
  - Lord of the Rings
  - Many others …
- nVidia + AMD/ATI Technologies push the performance of video cards on personal computers, tablets and mobiles.
- OpenGL + DirectX moving forward.
  - OpenGL 4.2 + DirectX 11

Graphics Cards

- 3D pipelines (nothing is fixed)
  - Optimised pipelines
  - Parallel Execution (16 vertices)
  - Programmable (Developers can implement their own shaders)
- Heterogeneous Computing
- General Purpose GPUs – massive data parallelisation

Current Research - Algorithms

- Global Illumination (Real-time) …
- Real-time Radiosity
- Real-time Raytracing
- Not rendering polygons any more ?!?!
- Ultimate Goal is (physically-based realism) …. The next goal is believable realism.