

Introducción a la Computación Gráfica

Ing. Gabriel Ávila, MSc.

Generalidades

Miércoles

- 8:00 a.m. a 11:00 a.m.
- Laboratorio de Ergonomía

Profesor

- Gabriel Ávila Buitrago.
- Ingeniero en Mecatrónica. MSc en Ingeniería de Sistemas y Computación.
- Correo: gabriel.avila@unimilitar.edu.co

Fechas importantes

Primera clase: 14 de agosto

Última clase: 20 de noviembre (Semana 16).

Parciales:

- 1er parcial: 4 de septiembre.
- 2do parcial: 16 de octubre.
- Examen final: 27 de noviembre.

Proyecto:

- Enunciado: 11 de septiembre.
- 1ra entrega: 23 de octubre.
- 2da entrega: 1 de mayo.
- 3ra entrega: 4 de diciembre.

Evaluación

	30%	30%	40%
Parciales	15%	15%	20%
Proyecto	5%	5%	10%
Otros	10%	10%	10%

Competencia global

El estudiante al terminar la asignatura de introducción a la computación gráfica, estará en capacidad de:

- **Analizar las condiciones esenciales que enmarcan un problema computacional relacionado con la generación de imágenes por computador.**
- **diseñar algoritmos que den solución a los planteamientos realizados.**

Así mismo, propone y diseña los niveles de interacción requeridos para que un usuario pueda interactuar con las aplicaciones.

¿De qué trata?

Proporcionar los conceptos fundamentales tanto teóricos como prácticos, en la generación de:

- **Primitivas gráficas.**
- **Transformaciones geométricas.**
- **Recorte y proyección de objetos.**
- **Interacción por medio de dispositivos E/S.**

para su posterior utilización en los cursos avanzados de Computación Gráfica y Simulación.

Herramientas



three.js

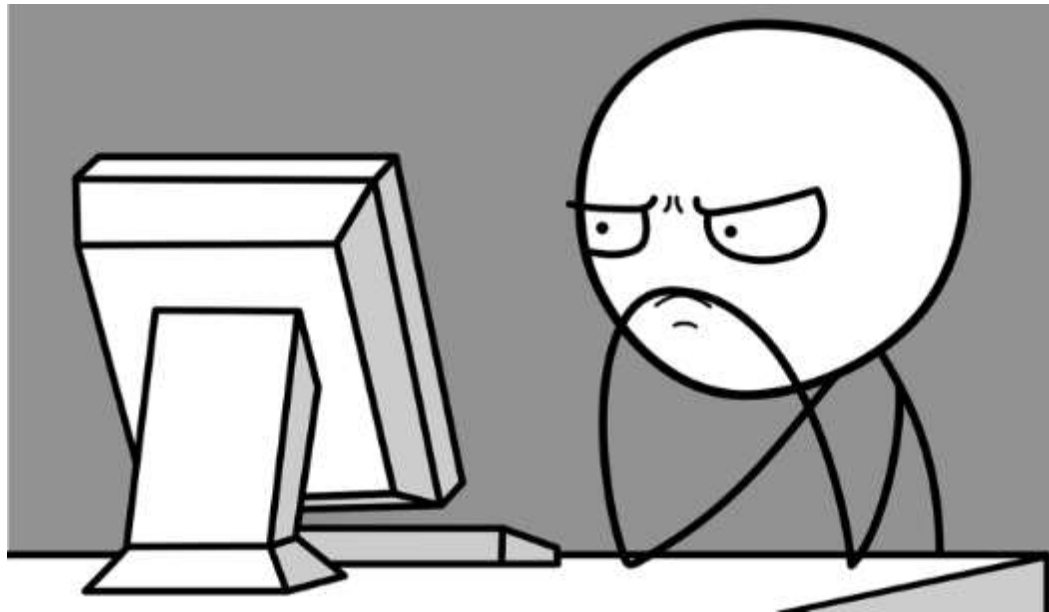


Bibliografía recomendada

- **Computer graphics with OpenGL.** Donald Hearn, M. Pauline Baker. Tercera Edicion, Editorial Prentice Hall, 2004. (biblioteca UMNG).
- **Computer graphics : C version.** Donald Hearn, M. Pauline Baker. Segunda Edicion, Editorial Prentice Hall, 1997. (biblioteca UMNG).
- **Computer graphics: principles and practice,** Foley, James D., el al.. Segunda Edicion, Editorial Addison Wesley, 2014. (biblioteca UMNG).
- **3D Computer graphics.** Alan Watt, Tercera Edicion, Editorial Addison Wesley, 2000. (biblioteca UMNG).
- **Interactive Computer Graphics, A top-down approach with OpenGL.** Edward Angel. Editorial Addison-Wesley, 2009. (biblioteca UMNG).
- **OpenGL: A Primer.** Edward Angel. Editorial: Addison-Wesley, 2008. (biblioteca UMNG).
- **OPENGL programming guide.** Woo et al. Addison-Wesley 1999. (biblioteca UMNG).
- **Mathematics for 3D game programming and computer graphics.** Eric Lengyel. 3ª ed. Boston, MA Cengage, 2012. (biblioteca UMNG).

Computación Gráfica en la Biblioteca UMNG Campus Cajicá ([link](#))

Ahora si... Comencemos!

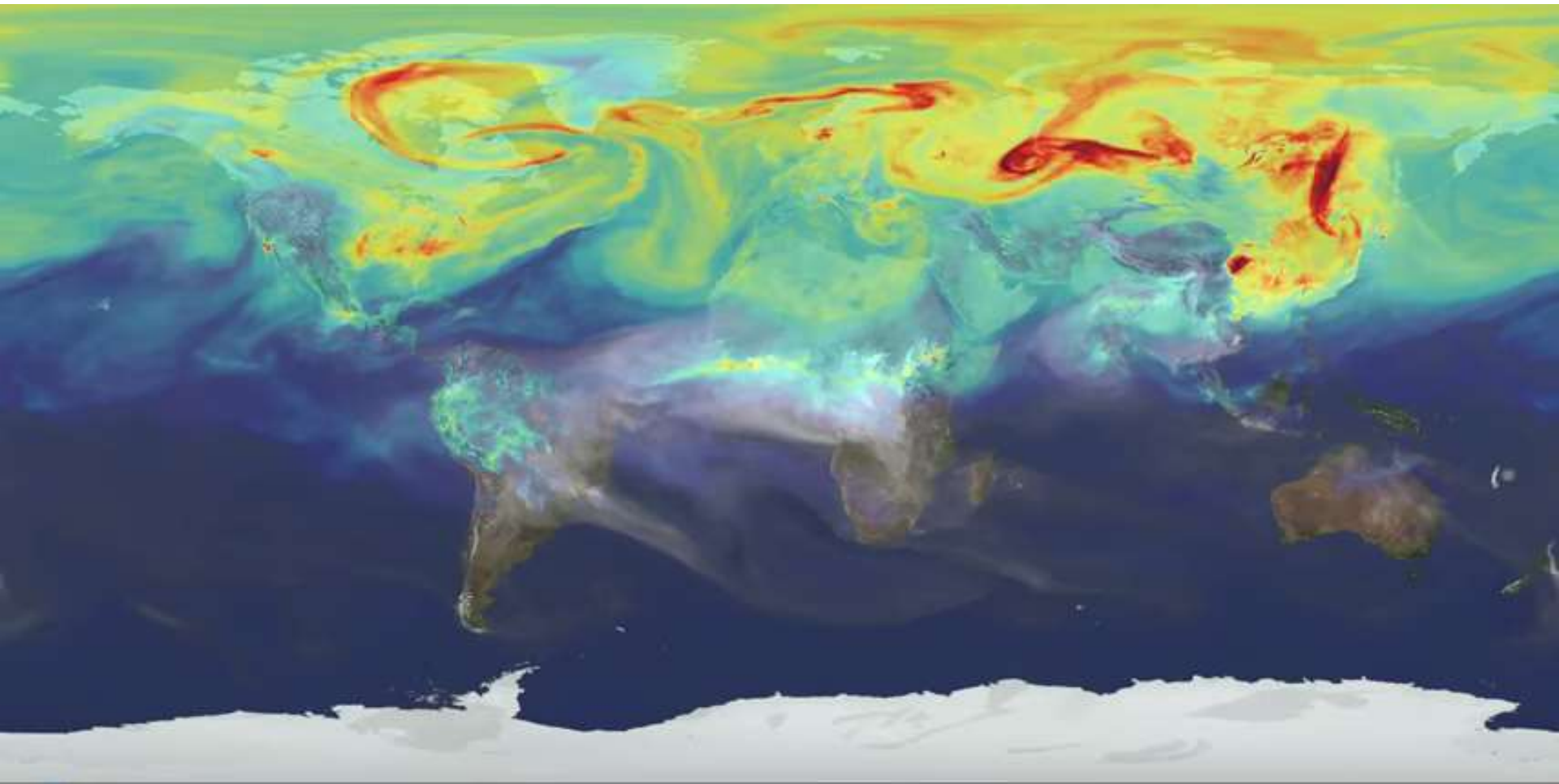


Tomado de: <https://medium.freecodecamp.org/make-your-hobby-harder-programming-is-difficult-thats-why-you-should-learn-it-e4627aee41a1>

¿Para qué la
computación gráfica?

¿Para qué la computación gráfica?

Visualización científica



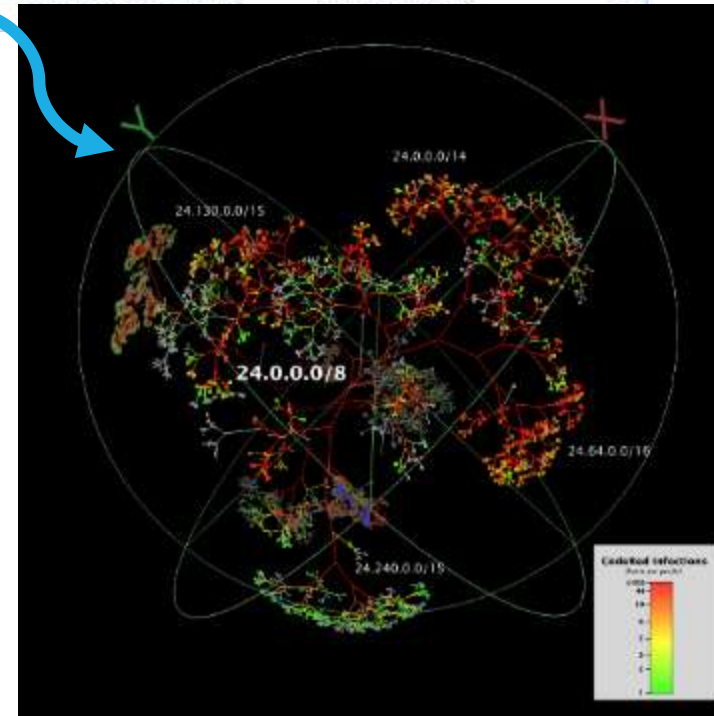
¿Para qué la computación gráfica?

Visualización científica

- Permite visualizar grandes cantidades de información.
- Utilizada para estudiar comportamientos complejos.
- Presentar tendencias y patrones de los datos, de manera comprensible.
- Facilita la toma de decisiones.

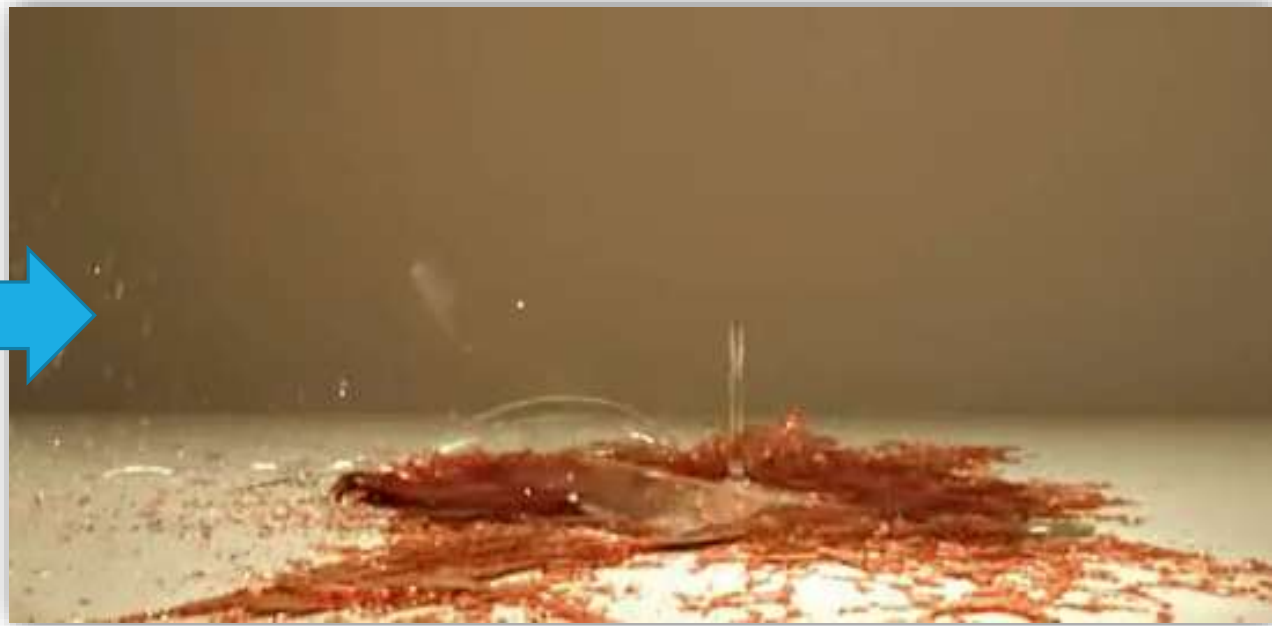
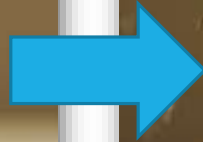
Tomado de: <https://www.webdesignerdepot.com/2009/06/50-great-examples-of-data-visualization/>

	A	B	C	D
1	DateTime	Longitude	Latitude	Altitude
281	4.5.15 7:18:37	6,97196211665869	51,41349929	134
282	4.5.15 7:18:41	6,97143933735788	51,41299747	136
283	4.5.15 7:18:45	6,97095804847777	51,4125539	137
284	4.5.15 7:18:49	6,97055228054523	51,41216489	137
285	4.5.15 7:18:53			



¿Para qué la computación gráfica?

Simulación



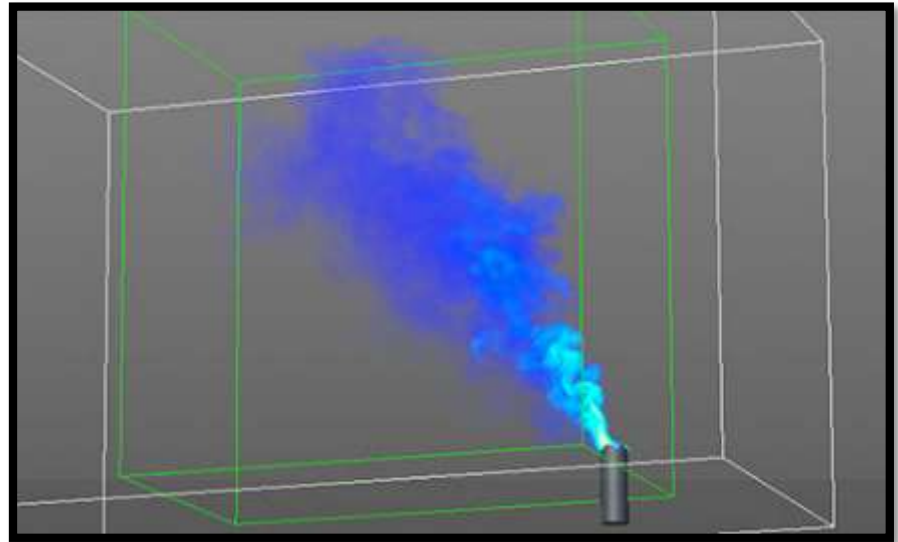
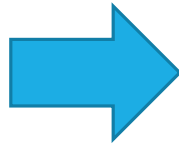
¿Para qué la computación gráfica?

Simulación

- Para realizar predicciones y análisis de comportamientos.
- Permite estudiar fenómenos naturales, partiendo de su comportamiento matemático.

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla) \mathbf{u} + \nu \nabla^2 \mathbf{u} + \mathbf{f}$$

$$\frac{\partial \rho}{\partial t} = -(\mathbf{u} \cdot \nabla) \rho + \kappa \nabla^2 \rho + S$$



Tomado de: <https://www.webdesignerdepot.com/2009/06/50-great-examples-of-data-visualization/>

¿Para qué la computación gráfica?

Videojuegos



- En el diseño de motores de videojuegos o la creación de nuevos tipos de interacción.
- Para aplicar en el funcionamiento de efectos especiales, interfaces, mejora de materiales, iluminación, etc.
- Permite generar todo tipo de gráficos, con diferente nivel de realismo.

Tomado de: <http://whatculture.com/gaming/17-best-video-game-covers-time>

¿Para qué la computación gráfica?

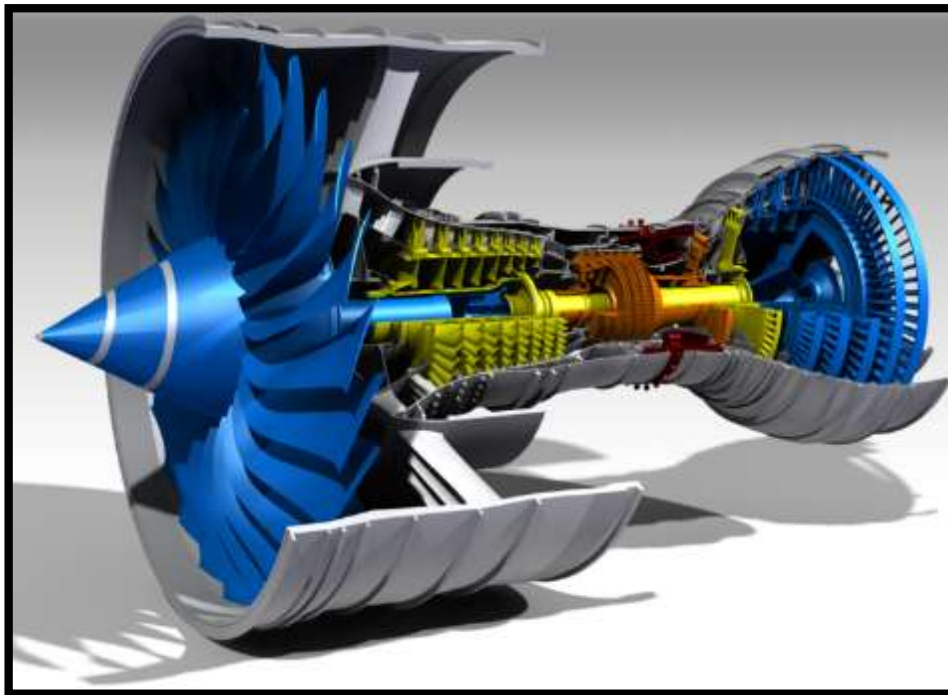
Diseño gráfico

- Permite producir imágenes para diversas áreas: arquitectura, mercadeo, etc.
- Diseño de escenas en 2D y 3D.

Tomado de: <https://simplydavidb.wordpress.com/2016/04/20/illustrator-example/#jp-carousel-233>

¿Para qué la computación gráfica?

Diseño Asistido por Computador



- Visualización y diseño de componentes mecánicos, eléctricos y de diferentes áreas.
- Facilita la estimación de materiales.
- Permite interactuar con los componentes de un sistema complejo.

¿Para qué la computación gráfica?

Realidad virtual y realidad aumentada



- Inmersión en entornos simulados.
- Dispositivos especializados para la interacción con objetos, bien sea en mundos virtuales o en el mundo real.

Computer Graphics (CG)

The study of creating, manipulating, and using visual images in the computer.

Some topics include:

- User Interface Design.
- Vector Graphics.
- 3D Modeling.
- Shaders.
- GPU design.
- Computer vision.
- Etc....[5]

Algo de historia

BASADA EN EL TIMELINE OF COMPUTER HISTORY [2]

Sus inicios – 1950s

Basados principalmente en los avances en ingeniería eléctrica, electrónica y en la televisión.

Avances en el uso de pantallas CRT para visualización e interacción.



SEAC Scanner - 1957



Robert Kirsch y su equipo (US National Bureau of Standards) diseñan un escaner de tambor rotatorio, que permite escanear imágenes por primera vez.

Inician los campos de Procesamiento de Imágenes y Reconocimiento de Patrones en Imágenes.

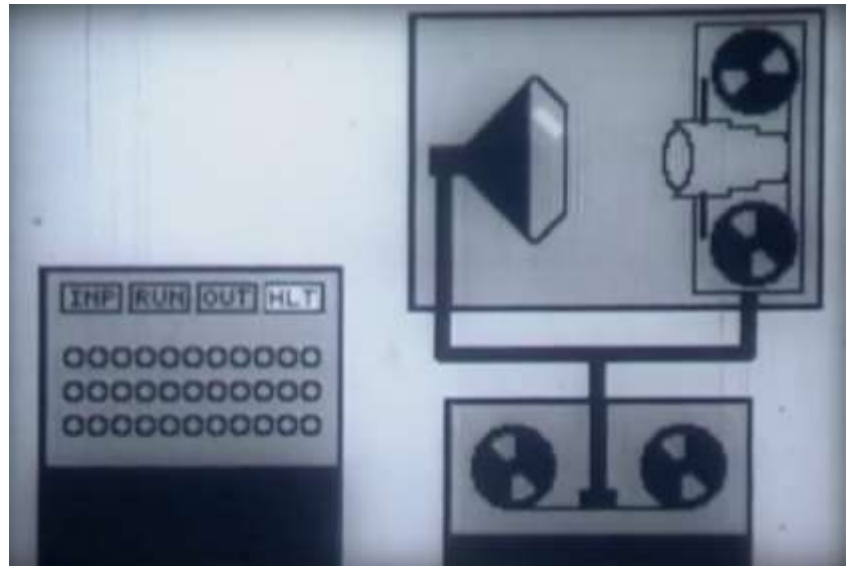


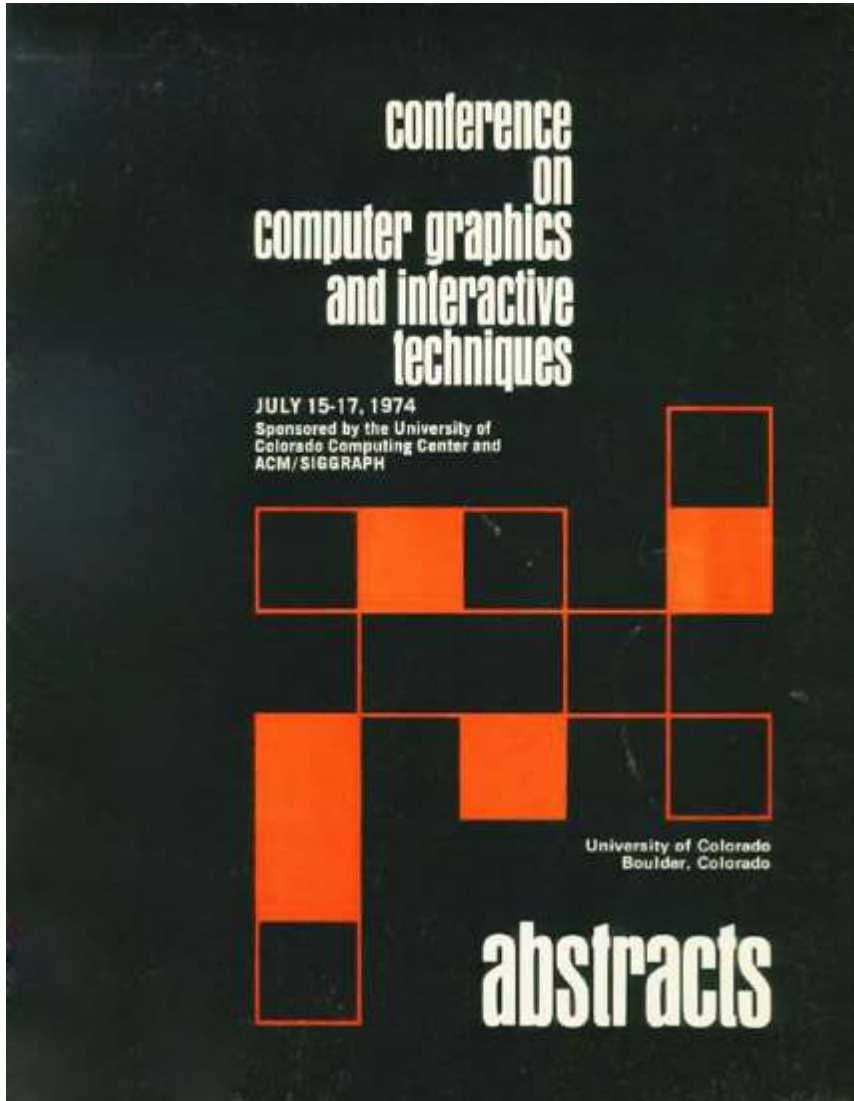
Beflix - 1961

Bell Labs investiga sobre
Computación Gráfica y Artística.

Beflix: Lenguaje especializado
para animación por computador.

Videos con imágenes de
252x184 pixeles.





Fundación de SIGGRAPH - 1969

Se crea la "Association for Computer Machinery Special Interest Group on Computer Graphics", uno de los grupos más influyentes en el área de la computación.

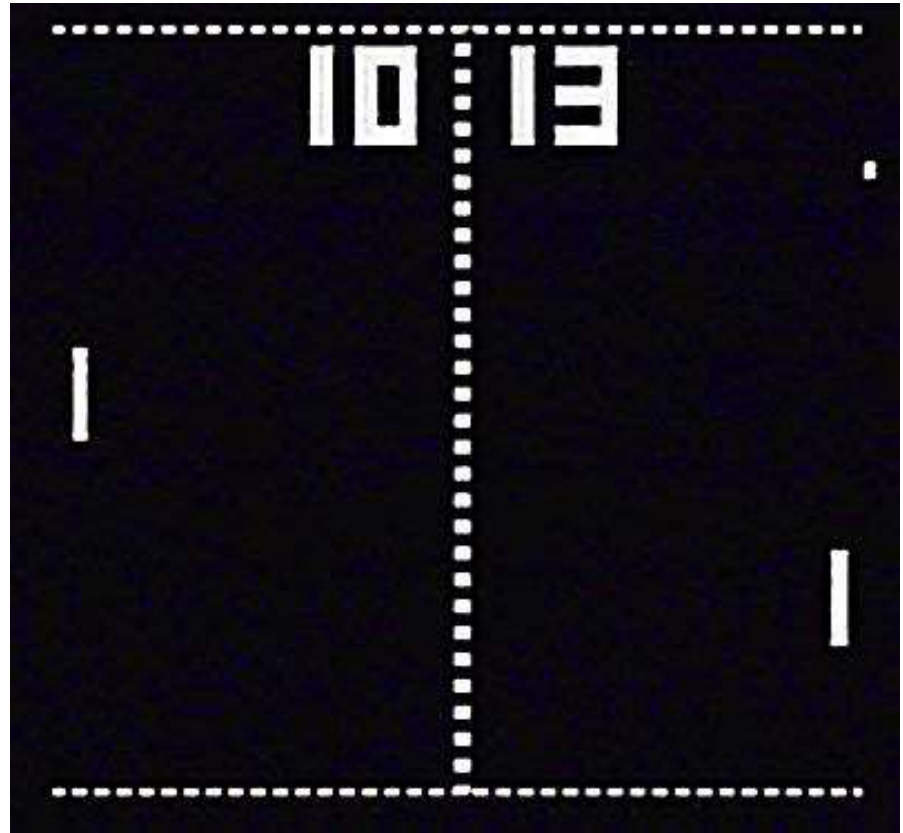
Anualmente se desarrollan conferencias en diferentes partes del mundo.

[Video](#) - 2018

Pong - 1972

Nolan Bushnell y Al Alcorn.

El inicio de la era de los videojuegos.



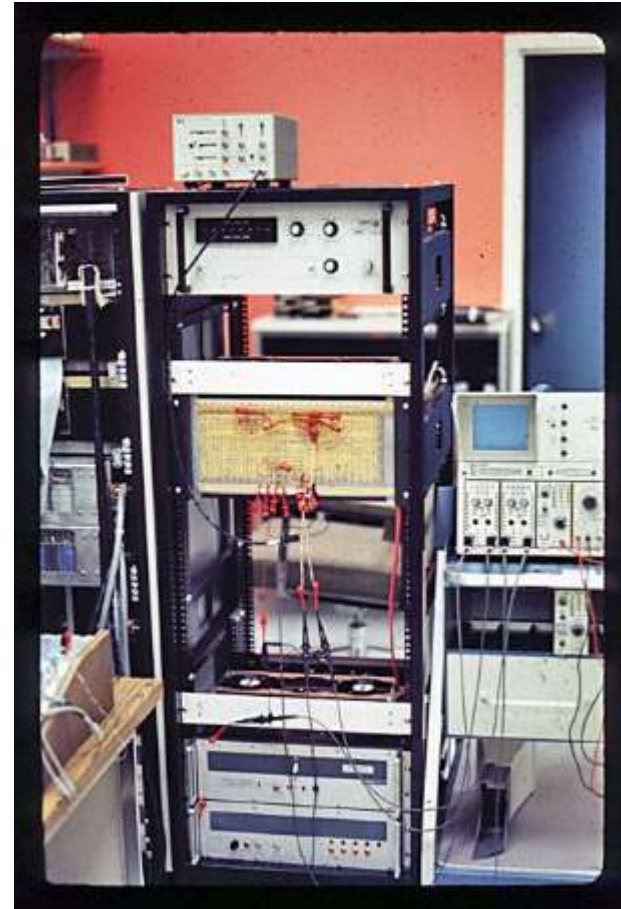
SuperPaint – 1972/74

XEROX Palo Alto Research Center (PARC).

Sistema de dibujo mediante un computador digital. Permitía crear animaciones sofisticadas, de hasta 16.7 millones de colores.

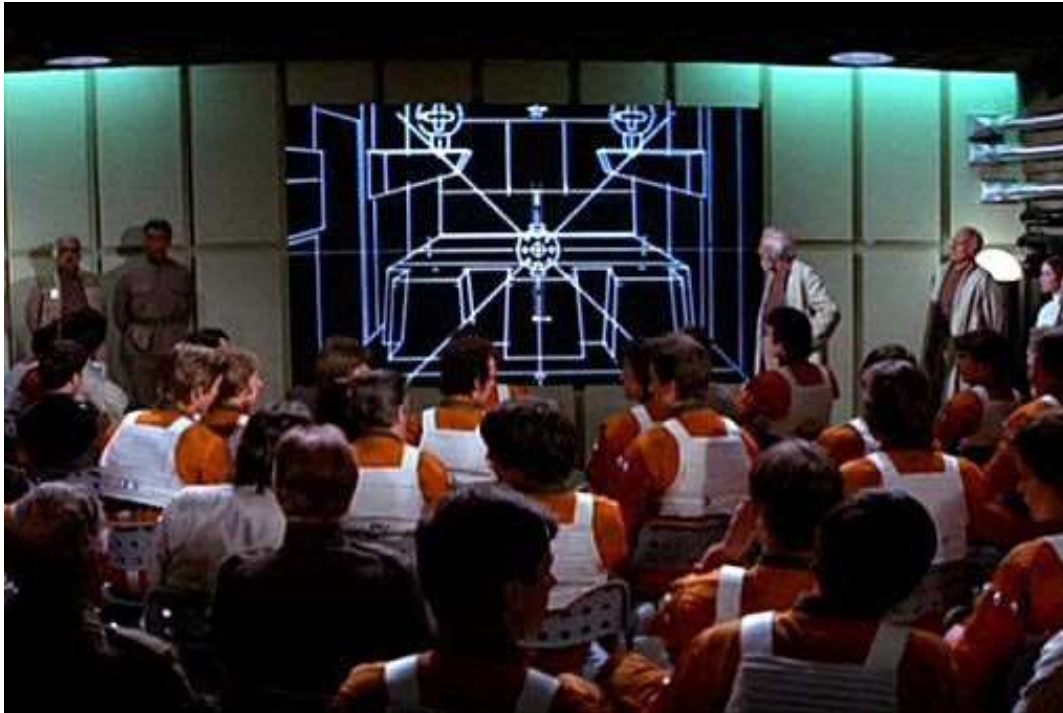
Características:

- Brochas ajustables.
- Magnificación de video.
- Uso de una tableta gráfica para dibujar.



Star Wars

Death Star Briefing - 1977



Industrial Light & Magic

Uno de los primeros usos de Wireframe-Animation en el cine.

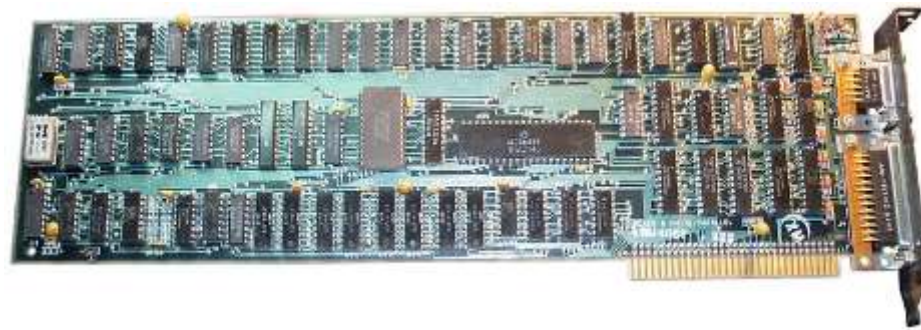
Hardware Gráfico – 1980s

Surgen las primeras tarjetas de video, dedicadas al manejo de gráficos 2D. Los fabricantes de hardware no seguían ningún estándar.

Los programadores debían dar soporte a cada pieza de hardware.

IBM Monochrome Display Adapter (1981):

Resolución de 720 x 350, monocromática.



Tomado de: <https://www.neweggbusiness.com/smartbuyer/components/22-game-changing-video-cards-1981-2015/>

Star Trek –1982



Lucasfilm

Basada en simulaciones realizadas por Jim Blinn para la NASA.

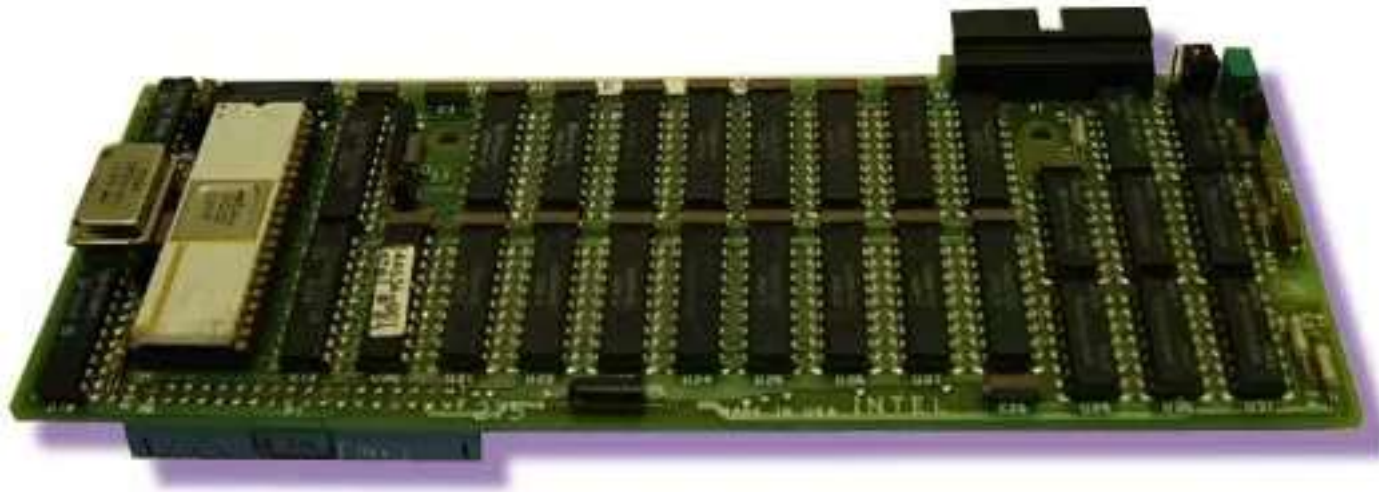
La secuencia se considera uno de los principales hitos en la animación por computador.

[Escena](#)

Hardware Gráfico – 1980s

Intel iSBX 275 Video Graphics Controller (1983):

Resolución de 256 x 256 con 3 bits de color (8 colores) o 512 x 512 monocromática.

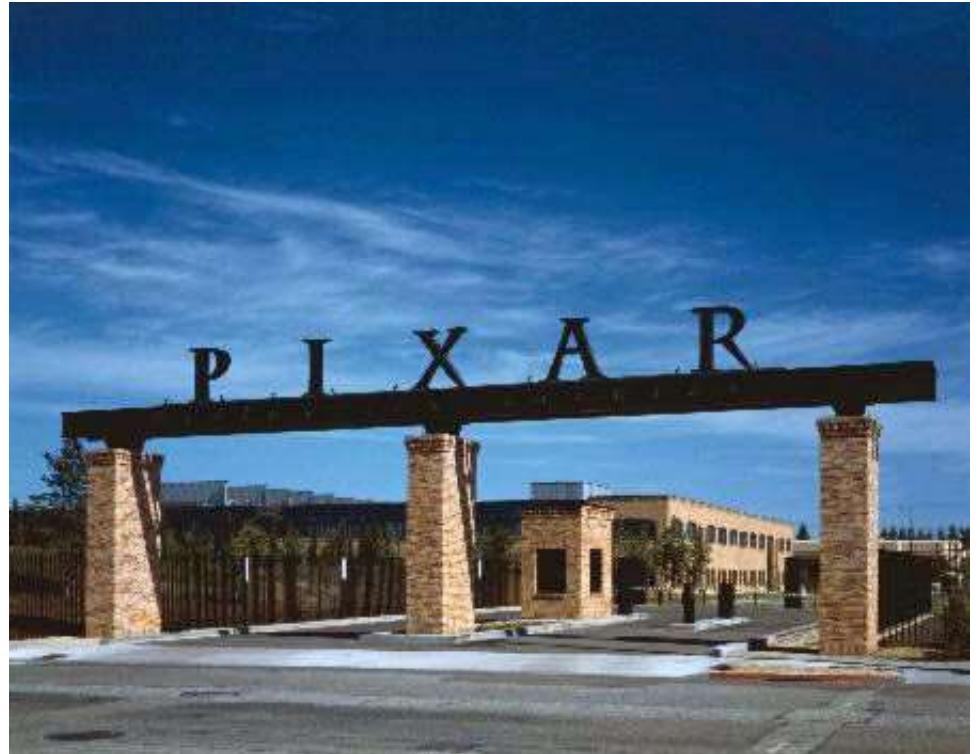


Tomado de: <https://www.neweggbusiness.com/smartbuyer/components/22-game-changing-video-cards-1981-2015/>

Fundación de Pixar - 1986

Se trataba originalmente del Grupo de Computación para Efectos Especiales de Lucasfilm.

Fue comprado por Steve Jobs en 1986 y vendido finalmente a Disney en 2006.



Hardware Gráfico – 1980s

ATI VGA Wonder (1988):

Resolución de 800 x 600 con 16 bits de color.



Tomado de: <https://www.neweggbusiness.com/smartbuyer/components/22-game-changing-video-cards-1981-2015/>

Finales de 1980s

Silicon Graphics (SGI) es líder en gráficos 3D con su librería IrisGL.

Su posición se debilita con la entrada de nuevos proveedores de hardware, lo que hace que decida convertir a IrisGL en un estándar abierto (precursor de OpenGL).

Basado en [2]

Inicio de 1990s

1990 – Comienza el desarrollo de OpenGL y la colaboración entre SGI y Microsoft.

1992 – Se completa OpenGL 1.0 y se crea la OpenGL Architecture Review Board (OpenGL ARB), un grupo de empresas que mantendrían y extenderían la especificación de OpenGL.

1995 – Se completa OpenGL 1.1 con soporte de texturas en GPU.

1995 – Microsoft desarrolla DirectX para manejo de gráficos en su SO.

1996 – Se hace pública la especificación de OpenGL.



Microsoft
DirectX

Basado en [2]

Hardware Gráfico – 1990s

Nvidia NV1 (1995)

Resolución 1200 x 600

Acelerador gráfico 2D y 3D, no soportaba DirectX 1.0 y fracasó (no permitía correr los juegos nuevos).



Hardware Gráfico – 1990s

3Dfx Voodoo 1 (1996)

Resolución 640 x 480 (3D)

Chip especializado en 3D.

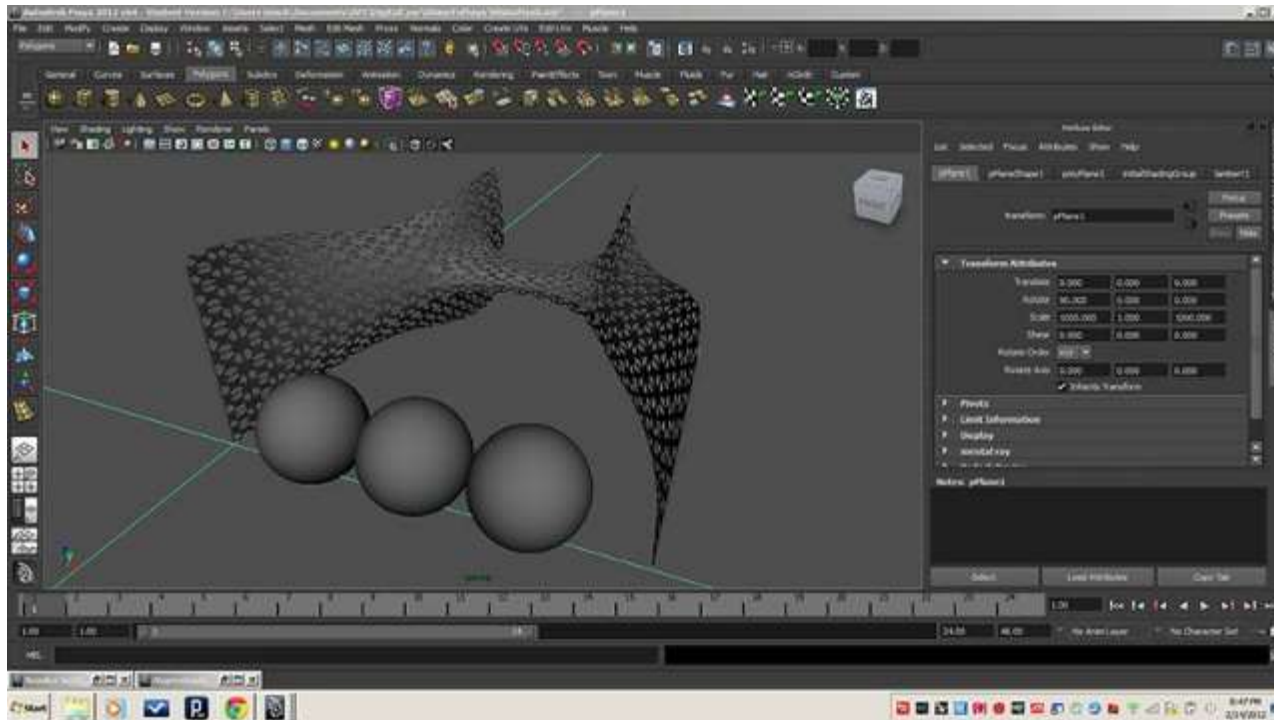
Dominó el mercado debido a su alto desempeño.

La empresa es vendida posteriormente a Nvidia.



Maya - 1998

Silicon Graphics Incorporate (SGI) desarrolla Maya como una herramienta de animación 3D de siguiente generación. Basado en sistemas previos creados por Wavefront y Alias.



Hardware Gráfico – 1990s



Nvidia GeForce 256 (1999)

Resolución 2046 x 1536

Debido al incremento en los requerimientos para computadores personales, Nvidia decide diseñar procesadores especializados para manipular gráficos.

Primeras GPU (Graphics Processing Unit), diseñadas para aliviar el procesamiento en la CPU.

Inicio de 2000s

2000 – OpenGL es liberado como código abierto.

1998 a 2003 – Se actualiza OpenGL a su versión 1.5 con capacidad para texturas volumétricas, multitexturas, sombreado por hardware, entre otras funcionalidades.

2004 – Se completa OpenGL 2.0 con la introducción de shaders (Pixel y Vertex), así como la definición del lenguaje GLSL (OpenGL Shading Language).

Basado en [2]

CUDA GPU - 2007



CUDA (Computer Unified Device Architecture) surge como un concepto que permite que las GPU realicen funciones reservadas previamente a las CPU.

Aprovecha el procesamiento multi hilos de las GPU, así como su escalabilidad.

Aplicaciones en la industria de juegos, biología computacional, criptografía, entre otros.

Finales de 2000s

2006 – Se completa OpenGL 2.1 con soporte para PBO (Pixel Buffer Objects).

2007-2009 – Se completa OpenGL 3.0 con mejora en el lenguaje de shaders y CUDA. Surge además OpenGL ES con soporte para dispositivos móviles y OpenCL con soporte para plataformas heterogéneas de CPU, GPU y otros procesadores.

Basado en [2]

2010s

2010 – Se completa OpenGL 4.0 con GLSL 4.0.

2010 – Se lanza la primera versión de Three.js para su uso con ActionScript, que posteriormente pasaría a Javascript.

2011 – Surge WebGL 1.0, un API de gráficos para Web, accesible desde el Canvas de HTML5. Basado en OpenGL ES 2.0.

2016 – Se completa OpenGL 4.6 (Versión actual).



three.js

Basado en [2] [3]

• • • •



Taller

Investigar sobre tarjetas gráficas en la actualidad.

Seleccionar una GPU en particular y buscar sus características (Fecha de lanzamiento, características generales, precio, etc.).

Agregar la información a la presentación compartida en el aula virtual.

Estudiar y revisar las características.

Dispositivos de Entrada y Salida

BASADO EN “3D USER INTERFACES: THEORY AND PRACTICE” DE BOWMAN (DISPONIBLE EN LA BIBLIOTECA UMNG)

[ENLACE](#)

Tarea

1. Ingresar a la página <http://www.opengl-tutorial.org/>
2. Seguir las instrucciones del tutorial inicial: <http://www.opengl-tutorial.org/es/beginners-tutorials/tutorial-1-opening-a-window/>
 - a. Descargar e instalar Visual Studio 2017
 - b. Descargar e instalar CMAKE
 - c. Descargar el código fuente de los tutoriales
 - d. Usar CMAKE para configurar e instalar el proyecto
 - e. Verificar que los archivos de tutorial se puedan abrir

Bibliografía

[1] Timeline of Computer

<http://www.computerhistory.org/timeline/graphics-games/>

[2] Introducción a la programación en OpenGL

[3] History of OpenGL. Tomado de:

https://www.khronos.org/opengl/wiki/History_of_OpenGL

[4] 22 Game Changing Video Cards, 1981-2015. Tomado de:

<https://www.neweggbusiness.com/smartbuyer/components/22-game-changing-video-cards-1981-2015/>

[5] Computer Graphics – Wikipedia. Tomado de:

https://en.wikipedia.org/wiki/Computer_graphics