

# Computación Gráfica en la Universidad Nacional del Sur

Castro Silvia, Delrieux Claudio, Silveti Andrea  
Computación Gráfica  
Dpto. de Ciencias de la Computación  
Universidad Nacional del Sur  
Avda. Alem 1253  
8000 - Bahía Blanca  
0291-4595135  
FAX 0291-4595136  
e-mail: [uscastro@criba.edu.ar](mailto:uscastro@criba.edu.ar)

## RESUMEN

---

Las currículas de los cursos de Computación Gráfica adecuados para los alumnos de Ciencias de la Computación e Ingeniería plantean un gran desafío al docente que debe diseñar sus contenidos. Computación Gráfica es una de las áreas de Ciencias de la Computación más dependientes de la tecnología y, sin duda, en estos últimos años ha habido avances increíbles en el campo tecnológico que la afectan directamente. Esto obliga a estructurar sus contenidos manteniendo el equilibrio entre los fundamentos y objetivos tradicionales y los cambios acelerados que se producen tanto en el contexto académico y cultural en los que se desenvuelve la enseñanza como en la actividad profesional que se desarrolla en el tema. Teniendo en cuenta estos factores, en la Universidad Nacional del Sur se desarrolló una metodología para la enseñanza de Computación Gráfica que no depende exclusivamente del diseño del contenido de los cursos.

---

## 1. ANATOMÍA DE UN CURSO INTRODUCTORIO

La mayoría de los cursos introductorios en Computación Gráfica [ 5 ] se basan en la enseñanza del pipeline gráfico tradicional: rasterización, construcción de modelos de objetos, transformaciones y *rendering*. En un cuatrimestre se cubren los siguientes tópicos:

- Introducción a la Computación Gráfica. Aplicaciones. Sistemas gráficos de *hardware* y de *software*.
- Algoritmos básicos de rasterización: líneas, círculos, relleno de polígonos. Generación de texto.
- Geometría básica en 2D. Transformaciones. Viewing en 2D.
- Geometría en 3D. Introducción al modelamiento de objetos en 3D. Transformaciones. Perspectiva y proyección.
- Introducción al problema de Cara Oculta.
- Introducción a los modelos de iluminación y sombreado.

El problema esencial en este caso es que, si se pretende que el estudiante escriba todo el software involucrado en el pipeline gráfico, el desarrollo de tal sistema de mediano porte le lleva casi todo el cuatrimestre y, recién al final del curso, está en condiciones de escribir algo realmente atractivo desde el punto de vista de Computación Gráfica, ya que los temas de interés no pueden cubrirse adecuadamente. Es claro que en la actualidad un curso tradicional de este tipo resulta totalmente insatisfactorio tanto desde el punto de vista de las motivaciones del estudiante como desde el punto de vista de las poderosas herramientas de hardware y de software con las que cuenta.

Otras propuestas para un curso introductorio, consisten en presentar sólo un survey de distintos tópicos de interés de Computación Gráfica. Aunque se cubran distintos aspectos de la programación en 3D y se manejan distintos conceptos interesantes de CG desde el punto de vista teórico se le dan al estudiante pocas posibilidades de programación. Esto resulta insatisfactorio ya que si bien ven todo lo que se puede hacer no se les brindan las herramientas necesarias para realizar dichas tareas.

Por último cabe considerar la posibilidad de programar aplicaciones en 3D desde el principio del desarrollo del curso. En este caso hay dos alternativas posibles: ray tracing [ 12, 7, 17] y un renderizador teniendo en cuenta el *pipeline* gráfico [ 2, 1, 4].

La primera de estas alternativas, parte del aprendizaje de los aspectos físicos de la interacción de la luz con los diversos materiales y su modelado computacional a través de algoritmos de *ray tracing*. En este caso, los alumnos utilizan desde el comienzo programas de *ray tracing* de distribución libre tales como el POV-RAY. Un curso con estas características propone, en general [ 12 ], la siguiente currícula en el orden en que se detalla a continuación:

- Principios físicos de la reflexión de la luz
- Ecuación general del *rendering*
- Modelos de iluminación global y local
- *Ray tracing*. Conceptos teóricos. Implementación de distintos algoritmos.
- Métodos aceleradores de *ray tracing*.
- Transformaciones de modelado en 2D y 3D.
- Algoritmos de rasterización.
- Transformación de *viewing*. Proyección. Perspectiva.
- Texturas. Mapeo.
- *Antialiasing*

Esta propuesta enfatiza un cambio desde la gráfica de 2D a la de 3D y resulta atractiva. Sin embargo, desde nuestro punto de vista presenta algunos inconvenientes. Por un lado, los *ray tracers* sólo pueden extenderse sencillamente con primitivas simples. Por otro lado, tanto las estaciones de trabajo como las PC utilizan la renderización a través del *pipeline* gráfico. Desde el punto de vista de la programación, los alumnos tienen un contacto superficial con la problemática del tema y pierden la oportunidad de desarrollar un sistema de mediano porte, que es la única manera de tomar contacto con la real dimensión del problema al no existir contacto con algunos aspectos teóricos fundamentales. Al mismo tiempo, el énfasis en el *ray tracing* hace perder de vista un objetivo de gran importancia, que es el desarrollo de aplicaciones interactivas ya que los *ray tracers* no permiten un *rendering* en tiempo real.

Hay también desventajas que provienen de la formación necesaria para poder aprovechar un curso de esta naturaleza. El curso parte de un énfasis en los aspectos físicos de la interacción de la luz con la materia.. Si bien esto es necesario para la comprensión de los modelos de iluminación utilizados en los problemas gráficos de gran realismo, su utilidad para aplicaciones normales es discutible. Además, para su adecuado desarrollo, los alumnos deben conocer óptica física, lo que no ocurre con los alumnos de la Licenciatura en Ciencias de la Computación.

La alternativa que tiene en cuenta el renderizador fue propuesta en [ 4 ] y posteriormente en [ 2 ]. En este caso, hemos visto cómo los estudiantes comienzan a programar aplicaciones no triviales. En el caso de nuestra propuesta anterior los contenidos del curso no diferían, en su introducción, del contenido tradicional. Como tema de importancia sustancial se agregaba el modelado con curvas y superficies, esencialmente los métodos de Bézier , B-Splines y  $\beta$ -Splines. Este tema insume un porcentaje considerable del tiempo de cursado, pero debido a que es uno de los temas de aplicación más relevantes de CG consideramos que no puede estar ausente en un curso introductorio. A esta altura del curso se visitan los laboratorios de Ingeniería donde los alumnos ven cómo se trabaja con estas herramientas en el diseño de estructuras. Hasta 1996 se utilizaba programación orientada a eventos, utilizando el Pascal para Windows como lenguaje de programación y algoritmos desarrollados por el grupo. Sin embargo, esto no era general. En [ 2 ] se propone OpenGL Application Programmers Interface (API) como una librería gráfica, debido a su aceptación en la comunidad gráfica y al soporte que tiene en la mayoría de las plataformas. Al mismo tiempo, en la Universidad Nacional del Sur también comenzó a utilizarse dicha API.

## 2. CURSOS INTRODUCTORIOS EN LA UNIVERSIDAD NACIONAL DEL SUR

En la Universidad Nacional del Sur se dictan cursos de Computación Gráfica en el Dpto. de Ciencias de la Computación y en el Dpto. de Ingeniería Eléctrica. Ambos cursos están a cargo de los integrantes del Grupo de Investigación en Computación Gráfica (GICG) de la Universidad Nacional del Sur que mantiene un constante

interés en las distintas propuestas de enseñanza, tratando de encontrar propuestas que se adecúen a los constantes cambios que se producen dentro de dicha área.

En el curso introductorio de Computación Gráfica dictado actualmente y desde 1997, se introduce a los alumnos directamente en el mundo 3D, siguiendo el pipeline gráfico, pero utilizando elementos de una API gráfica ampliamente aceptada. A mediados de cuatrimestre, los alumnos están en condiciones de escribir un sistema gráfico interactivo de mediano porte, tomando contacto con la problemática de un sistema interactivo. El desarrollo de un sistema gráfico les permite un entendimiento profundo de conceptos básicos en Computación Gráfica que se presentan siguiendo el pipeline gráfico directamente en 3D. Para disminuir el costo en tiempo que tal desarrollo implica, la presentación de los proyectos que se realizan durante el desarrollo del curso se realiza en C, utilizando OpenGL como API gráfica. De este modo, el alumno cuenta con primitivas gráficas más especializadas que sólo las soportadas en los lenguajes de programación. El uso de esta API facilita grandemente la enseñanza de conceptos clásicos de Computación Gráfica y la implementación de los algoritmos permitiendo ingresar rápidamente al mundo de 3D y tocar durante este primer curso diversos tópicos avanzados que son de gran interés para los alumnos. Paralelamente, y a medida que se hace necesario se retoman en distintos momentos tópicos necesarios e importantes tales como rasterización.

Además, debido a las inquietudes cada vez más exigentes de los alumnos, sobre el final del curso, y como trabajo final del mismo, los alumnos eligen un tema y realizan una monografía y algunas veces desarrollan una aplicación en un área que refleja la investigación actual en el tema.

Actualmente la currícula de Computación Gráfica es la que se detalla a continuación:

- Introducción a los Sistema Gráficos de hardware y de Software. Introducción a OpenGL.
- Geometría básica en 3D. Transformaciones. Espacio del objeto y de la imagen. Perspectiva y proyección. Clipping, windowing y viewport.
- Cara oculta.
- Modelos de iluminación local. Sombreado.
- Modelamiento mediante curvas y superficies.
- Modelos de iluminación no locales. Ray Tracing. Radiosidad.
- Introducción a los modelos procedimentales
- Introducción a la visualización científica
- Introducción a los sistemas de animación

A mitad del cuatrimestre los alumnos pueden escribir programas moderadamente complejos e interactivos, habiendo entendido cabalmente cómo funciona el pipeline gráfico. A esta altura se discuten los temas de iluminación y sombreado, que son incorporados posteriormente a sus sistemas. En tanto esto es realizado, se introducen temas de modelamiento avanzado mediante curvas y superficies, herramientas que incorporan como último trabajo práctico antes del trabajo final. Mientras realizan este último trabajo, en teoría se los introduce a distintas áreas de interés en Computación Gráfica, sobre alguna de las cuales trabajarán en su proyecto final. Este proyecto final implica la lectura profunda de un paper de actualidad en el tema. Esto es sin duda de una dificultad considerable y muchas veces es la única oportunidad en que los alumnos toman contacto directo con material de investigación. El proceso de selección comienza con la elección del tema y, en base al grado de dificultad del tema elegido y a cómo se desempeñaron los alumnos durante el cuatrimestre, la cátedra selecciona el trabajo de investigación adecuado. De acuerdo a las características y dificultades del mismo, puede requerirse o no la implementación de las ideas o un conjunto reducido de ellas.

### **3. CURSOS AVANZADOS EN COMPUTACIÓN GRÁFICA**

El contenido del curso introductorio es ambicioso. Sin embargo, muchas áreas importantes de Computación Gráfica quedan sin duda fuera del mismo. Por otro lado si bien hay temas que se introducen, no se cubren con la profundidad adecuada. Es por esto que, para aquellos que desean completar su formación en Computación Gráfica se diseñaron distintos cursos optativos cuyos contenidos se actualizan año a año para cubrir los avances en este campo de aplicación.

El curso *Tópicos Avanzados en Computación Gráfica* fue dictado por primera vez en 1994 para un grupo reducido de alumnos. El programa del mismo abarca:

- **Tópicos avanzados en aproximación de superficies.** Modelamiento de superficies de forma libre con sistemas de coordenadas auxiliares. Generalización de las superficies de Bézier para el modelamiento de topologías arbitrarias. Splines triangulares. NURBS.
- **Modelos de iluminación y sombreado.** Modelos de iluminación avanzados. Mapas de texturas, mapas de desplazamiento y mapas de normales. Espacios de color. Paletas estáticas, dinámicas y adaptivas.
- **Ray Tracing.** Ray tracing como solución al problema general de iluminación y sombreado. Métodos generales de aceleración del algoritmo de ray tracing.
- **Animación.** Técnicas básicas de animación convencional. Aplicación en Computación Gráfica. Animación con key-frames. Interpolación con in-betweening automático.
- **Síntesis de imágenes con conjuntos fractales.** Simulación de fenómenos naturales con fractales no determinísticos y algoritmos estocásticos. Simulación de plantas, árboles y objetos vivos con fractales autosimilares, gramáticas y sistemas de reescritura.

Un segundo curso, orientado a las aplicaciones, es el de Visualización. El objetivo del curso es brindar una introducción al tema, es decir, a la representación gráfica computacional de datos que, por una u otra razón no pueden representarse en términos de gráficos convencionales. El resultado que se espera de la visualización de estos datos no es meramente cuantitativo ya que no se busca necesariamente la representación fiel de valores, sino cualitativo, es decir que se persigue un entendimiento global de determinadas propiedades de los datos. Los temas tratados en este curso son:

- **Introducción a la Visualización** Objetivos. Breve historia. Distintas ramas y aplicaciones representativas.
- **Color y Sonido.** Concepto de color. Percepción humana del color. Espacios de color. Concepto de sonido. El uso del color y del sonido en Visualización.
- **Datos.** Fuentes de obtención de los datos para una visualización. Clasificación de los datos. Atributos.
- **El Pipeline de Visualización.** Pautas básicas de representación y visualización de datos y procesos. Topología del pipeline. Ejecución del pipeline. Modelos conceptuales para sistemas de Visualización.
- **Visualización Científica.** Visualización de Datos Volumétricos. Visualización de datos sobre Superficies. Visualización de Flujos.
- **Visualización de Software.** Animación de Algoritmos. Visualización de Programas. Aplicaciones de visualización de software a dominios específicos de programas.
- **Visualización de Información**

Los alumnos de la Licenciatura o de la Ingeniería que hayan tomado el curso introductorio y por lo menos uno de los cursos avanzados podrán encarar la tesis final de carrera en Computación Gráfica. Entre algunos de los temas de actualidad que pueden ser actualmente temas de tesis, podemos citar:

- Edición de curvas Multirresolución.
- Edición de imágenes
- Compresión de imágenes
- Compresión de objetos en 3D
- Warping y morphing en 2D y en 3D
- Modelamiento multirresolución
- Visualización colaborativa

- Visualización en Internet
- Texturas procedurales
- El arte en Computación Gráfica
- Realidad Virtual

Debido a la sólida formación que ha adquirido el GICG se está estudiando la implementación de un posgrado en Computación Gráfica. Sin duda esto es de gran relevancia, ya que actualmente no tenemos conocimiento de otros grupos consolidados en esta rama de conocimientos a nivel nacional.

#### 4. CONCLUSIONES

Presentamos cuáles son las alternativas sobre el dictado de distintas asignaturas de Computación Gráfica en la Universidad Nacional del Sur. Con respecto a un curso introductorio, se propone un equilibrio entre la currícula tradicional y distintas propuestas modernas, tratando de balancear y compensar las ventajas y desventajas observadas en cada una.

Con respecto a los cursos optativos se presentan los contenidos de las distintas alternativas que posibilitan una formación más completa y un acercamiento a los temas de investigación. De este modo se tiende a la implementación de un posgrado en Computación Gráfica.

#### 5. REFERENCIAS

- [ 1 ] Angel, E., *Interactive Computer Graphics: A top-down approach with OpenGL*, Addison-Wesley, Reading, Massachusetts, segunda edición, 1997.
- [ 2 ] Angel, E., *Teaching a Three-Dimensional Computer Graphics Class Using OpenGL*, Computer Graphics, Vol 31(3), 1997.
- [ 3 ] Bartels, R., Beatty, J., Barsky, B. , *An Introduction to Splines for Use in Computer Graphics and Geometric Modelling*, Springer Verlag, New York, 1987.
- [ 4 ] Delrieux,C., Castro,S., Silvetti,A., *Enseñanza Introductoria y Avanzada en Computación Gráfica*, Tercer Ateneo de Profesores Universitarios de Computación, Bahía Blanca, 1995.
- [ 5 ] Foley,J., Van Dam,A., *Fundamentals of Interactive Computer Graphics*, Addison-Wesley, Reading, Massachusetts, segunda edición, 1992.
- [ 6 ] Gallagher, R., *Computer Visualization: Graphics Techniques for Sc. And Eng. Analysis* , 1996
- [ 7 ] Grissom,S., Bressenham,J., Kubitz,B., Owen,S., Schweitzer,D., *Approach to Teaching Computer Graphics*, ACM SIGCSE, ACM Press, Nashville, TN, 1995.
- [ 8 ] Hearn,D., Baker,P., *Computer Graphic*, Prentice Hall,New Jersey, Second Edition, 1997.
- [ 9 ] Kelly, P., Keller, M., *Visual Cues: Practical Data Visualization*, IEEE Computer Society Press, 1992.
- [ 10 ] Nielson, G., Shriver, B. *Visualization in Scientific Computing*, IEEE Computer Society, 1990.
- [ 11 ] Nielson, G., Hagen, H., Müller, H. *Scientific Visualization: Overviews, Methodologies and Techniques*, IEEE Computer Society, 1997.
- [ 12 ] Owen,A., Larrondo-Petrie,M., Laxer,C., *Computer Graphics Curriculum: Time for a Change?*, Computer Graphics, Vol 28(3), 1994.
- [ 13 ] Schroeder, W., Martin, K., Lorensen, B., *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*, Prentice Hall PTR, 1996.

[ 14 ] Vince, J., *3D Computer Animation*, Addison-Wesley Publishers, Gran Bretaña, 1992.

[ 15 ] Watt, A., *3D Computer Graphics*, Addison-Wesley Publishing Company, 1993.

[ 16 ] Watt, A., Watt, M., *Advanced Animation and Rendering Techniques: Theory and Practice*, Addison-Wesley Publishing Company, 1993.

[ 17 ] Xiang,Z., *A Nontraditional Computer Graphics Course for Computer Science Students*, Computer Graphics, Vol 28(3), 1994.