

Enseñanza Introductoria y Avanzada en Computación Gráfica

Claudio Delrieux¹, Silvia Castro², y Andrea Silvetti³

ICIC - Instituto de Ciencias e Ingeniería de Computación
Grupo de Investigación en Computación Gráfica
Universidad Nacional del Sur
Alem 1253 - (8000) Bahía Blanca - ARGENTINA
e-mail: {usdelrie,uscastro,silvetti}@criba.edu.ar

Resumen:

Se presenta una discusión de los contenidos, requisitos y metodología de enseñanza en computación gráfica en varios niveles. El conjunto de cambios tecnológicos ocurridos en la última década determina que la enseñanza introductoria en computación gráfica deba replantearse sustancialmente. Recientemente, en el seno del Comité de Educación de la ACM SIGGRAPH se propuso una modificación sustancial a la currícula de los cursos de computación gráfica, enfatizando un acercamiento basado en los principios físicos de la luz. La propuesta recomienda utilizar *software* comercial y programas de distribución libre para plataformas convencionales (PC) para acelerar la comprensión de los principios fundamentales de la disciplina. En este trabajo se discuten los problemas que acarrea este último enfoque, y se propone una metodología que se considera más adecuada para la enseñanza introductoria y avanzada en computación gráfica.

¹Departamento de Ingeniería Eléctrica

²Departamento de Ciencias de la Computación

³Departamento de Ciencias de la Computación y becaria de Iniciación del CONICET

Enseñanza Introdutoria y Avanzada en Computación Gráfica

1 Introducción

Los cambios tecnológicos, la evolución de las plataformas de desarrollo, el abaratamiento del *hardware*, la disponibilidad de programas comerciales con aplicaciones gráficas avanzadas, y la proliferación de sistemas gráficos de distribución libre, determinan en conjunto que un curso introductorio de computación gráfica estructurado en forma clásica (siguiendo los libros tradicionales como [26, 17, 14, 41]) pueda tornarse obsoleto o inadecuado. Pero al mismo tiempo, un cambio radical en la currícula de dicho curso puede presentar inconvenientes insalvables. Depender de sistemas comerciales o programas de distribución libre puede imponer limitaciones *a priori* a las posibilidades exploratorias y a la capacidad de extensión. Además es un factor que limita la creatividad, distrae el interés en los aspectos teóricos, y disminuye el énfasis en la programación. Por lo tanto, el contenido debe reflejar el precario y cambiante equilibrio existente entre los fundamentos y objetivos tradicionales de un curso introductorio en computación gráfica, y los cambios acelerados en el contexto académico y cultural en que se desenvuelve la enseñanza y la actividad profesional en el tema.

Al mismo tiempo, los usos avanzados de la computación gráfica han producido una gradual pero profunda revolución en la manera en que se utiliza la computadora en los contextos científicos y tecnológicos. Tal es el caso, por ejemplo, de las disciplinas centradas en lo que hoy se conoce como *visualización científica* [27, 21], las cuales constituyen en la actualidad un eje fundamental del cambio metodológico de la investigación en áreas del conocimiento cada vez más diversas [35, 7]. Por lo tanto, es necesario contemplar la preparación de docentes y profesionales en ámbitos en los que es indispensable el uso de tópicos avanzados en computación gráfica a través de materias optativas y cursos de posgrado. Esto plantea, a su vez, nuevos problemas en los contenidos de las materias. Este conjunto de circunstancias llevó a determinar una metodología para la enseñanza introductorio y avanzada en computación gráfica, la cual no depende exclusivamente del diseño de las currículas.

2 Organización de un curso introductorio en Computación Gráfica

Presentamos en esta sección la descripción, objetivos, requisitos previos y organización de un curso introductorio en Computación Gráfica. Se discuten las desventajas que hacen que el mismo sea inadecuado a la luz de los cambios mencionados en la introducción. En la siguiente sección se presentan enfoques alternativos surgidos en determinados ámbitos académicos, en función de la disponibilidad de medios y de la mejor preparación y motivación de los estudiantes. En la sección 4 se presenta una propuesta para la organización de un curso introductorio en el tema. La misma se mantiene cercana al enfoque tradicional pero incorpora los cambios tecnológicos como facilidades en la plataforma

de desarrollo de programación de los trabajos prácticos. Esto permite incorporar un conjunto de herramientas que facilitan notablemente a los alumnos la implementación de programas. El ahorro en tiempo de desarrollo permite incorporar un acercamiento directo a temas teóricos a elección, ampliando así el alcance del curso.

Un curso introductorio de computación gráfica está orientado a brindar un conjunto de conceptos y técnicas a los estudiantes de una Licenciatura en Ciencias de la Computación o de una Ingeniería. Los objetivos básicos incluyen una adecuada cobertura de la organización del *hardware* y *software* de base gráficos, conceptos de representación y manipulación geométrica en 2D y 3D, presentación de información univaluada y bivaluada, y una introducción a temas avanzados como cara oculta, modelos de iluminación, curvas y superficies algebraicas y modelos procedimentales. Los requisitos mínimos para afrontar un curso de estas características consisten en un buen dominio de geometría analítica, algoritmos y estructuras de datos, y un manejo disciplinado de una plataforma de desarrollo de programas.

La organización tradicional de un curso introductorio (de duración cuatrimestral) en Computación Gráfica [26, 17, 14] cubre los siguientes tópicos:

- Introducción a los sistemas de *hardware* y *software* gráficos de una computadora.
- Desarrollo de programas gráficos de base (líneas y círculos).
- Geometría básica en 2D. Transformaciones.
- Espacios de objeto y de imagen. *Clipping*, *windowing* y *viewport*.
- Geometría en 3D. Transformaciones. Perspectiva.
- Introducción al problema de línea y cara oculta.
- Introducción a los modelos de iluminación y sombreado.

Un curso de estas características cubre en un mínimo los objetivos planteados, esencialmente porque el tiempo empleado en el desarrollo de los trabajos prácticos no permite afrontar los temas más avanzados. En su mayor parte, los temas de interés o de gran aplicabilidad (como por ejemplo el modelado con curvas y superficies algebraicas) no pueden ser adecuadamente cubiertos.

Este era, sin embargo, el mejor contenido que se podía adecuar a un contexto como el existente hace 10 años, en el cual tanto las expectativas como las posibilidades de equipamiento eran mucho menores. En la actualidad, tanto la motivación de los estudiantes como el manejo de computación que tienen en general, hace que un curso tradicional como el mencionado resulte totalmente insatisfactorio.

3 Tendencias actuales

Al mismo tiempo que maduró la disciplina y se abarató el *hardware*, es posible observar un mayor manejo y experiencia en el uso de computadoras que tiene un estudiante

promedio. Por otra parte, muchos estudiantes poseen equipos de prestaciones mucho mayores que los que la Universidad les puede ofrecer. Al mismo tiempo, se popularizan cada vez más los libros de texto con diskettes y CD-ROM con demos, programas gráficos e imágenes. Estos cambios en el contexto de los estudiantes hacen que las expectativas puestas en un curso sean notablemente más sofisticadas.

Recientemente un grupo de trabajo del Comité de Educación de la ACM SIGGRAPH propuso una modificación sustancial a la currícula de los cursos de computación gráfica [38, 29, 30]. Básicamente la propuesta consiste en utilizar *software* comercial y programas de distribución libre para plataformas convencionales (PC) para acelerar la comprensión de los principios fundamentales de la disciplina. Los estudiantes disponen, además, del apoyo de artistas profesionales para ser orientados en los aspectos visuales y estéticos de sus trabajos.

El curso enfatiza esencialmente la producción de gráficos en 3D con realismo. Para ello, el punto de partida consiste en el 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*. Más adelante, o en cursos avanzados, los estudiantes toman contacto con el enfoque tradicional *scan-line* de la computación gráfica. La secuencia propuesta en [30] (y por otros educadores en otros foros [19, 28, 43]) es la siguiente:

- Principios físicos de la reflexión de la luz.
- La ecuación del *rendering*.
- Modelos globales de iluminación.
- Modelos sencillos de iluminación local (Phong).
- Teoría e implementación de un algoritmo de *ray tracing*.
- Métodos para acelerar el *ray tracing*.
- Transformaciones en 2D y 3D.
- Algoritmos *scan-line* para rectas y círculos.
- *Clipping* en 2D. Conversión *scan* de polígonos.
- Cara oculta.
- Mapeo de texturas y desplazamientos.
- Curvas y superficies paramétricas.
- Anti-*aliasing*.
- Modelos procedimentales (fractales, sistemas iterados).

Los alumnos de un curso de estas características utilizan desde el comienzo programas de *ray tracing* de distribución libre como el POV-RAY, para luego reprogramarlos o modificarlos según sus propios intereses. Una vez manejados los conceptos de la

primera mitad del curso, los alumnos aprenden el funcionamiento teórico de los algoritmos *scan-line* tradicionales utilizando programas comerciales como el Renderman de PIXAR. Como es posible observar, el contenido del curso es bastante ambicioso, dado que brinda la posibilidad de manejar los conceptos avanzados desde un comienzo, y motiva a los estudiantes dado que producen imágenes con una calidad comparable a la de los medios comerciales.

Sin embargo, desde nuestro punto de vista, esta orientación tiene desventajas profundas que es necesario analizar. La más importante es que, a nuestro entender, el uso de sistemas preexistentes tiende a generar un contacto superficial con la problemática del tema, produciendo *usuarios* más que solucionadores de problemas. El desarrollo de programas propios es la única manera de tomar contacto con la real dimensión del problema. Existe una falta de contacto con algunos aspectos teóricos fundamentales. Al mismo tiempo, se limita la creatividad, puesto que si bien ofrece un panorama amplio y rápidamente accesible, las posibilidades de graficación están circunscriptas a una solución particular. Sin un conocimiento *a priori*, las limitaciones de un producto pueden confundirse con las limitaciones teóricas.

Otras desventajas provienen de la formación necesaria para poder aprovechar un curso de esta naturaleza. Desde el comienzo se insiste en los aspectos físicos de la interacción de la luz con la materia. Si bien esto es necesario para la real comprensión de los modelos de iluminación utilizados en los programas gráficos de gran realismo, su utilidad para las aplicaciones normales es discutible. Lo mismo sucede con la temática relacionada con el anti-*aliasing* y los modelos de muestreo estocástico. Una comprensión adecuada de esta temática requiere que los estudiantes conozcan óptica física, probabilidad, análisis de variable compleja y procesamiento de señales discretas. Es decir, el curso pasa a ser mucho más exigente en sus correlativas.

Desde el punto de vista de la programación, los estudiantes pierden la oportunidad de desarrollar un sistema de mediano porte, lo cual puede ser un objetivo en sí mismo para su formación –al margen de que el sistema implementado sea de computación gráfica u otra disciplina. El tiempo “ahorrado” se invierte en implementaciones que tienden al realismo en los modelos de iluminación pero que dejan de lado otras temáticas dentro de la computación gráfica que son igualmente importantes y cuya aplicación en otras áreas es mayor. Al mismo tiempo, el énfasis en el *ray tracing* hace perder de vista un objetivo esencial, que es el de desarrollar aplicaciones interactivas.

4 Una propuesta alternativa

El Grupo de Investigación en Computación Gráfica (GICG) de la UNS mantiene un constante interés en las distintas propuestas de enseñanza, tratando de encontrar una solución adecuada dentro de los cambios mencionados más arriba. En cada uno de los enfoques presentados se observan ventajas y desventajas, con las que es posible elaborar una síntesis. El desarrollo de un sistema gráfico permite al alumno un entendimiento profundo del tema. Sin embargo, es necesario buscar una forma de disminuir el costo en tiempo que tal desarrollo implica, agilizando al máximo el desarrollo de los programas interactivos por parte de los alumnos.

Con respecto a las inquietudes cada vez más exigentes de los alumnos, sobre el final del curso, cada uno elige un *paper* que refleje investigación actual en revistas de primer nivel (IEEE Computer Graphics and Applications, ACM Transactions on Graphics, ACM SIGGRAPH Proceedings) para su exposición en clase. Muchas veces de dichas discusiones surgen resultados que realimentan el contenido de la materia para los años siguientes. De esa manera los alumnos se sienten capacitados para el desarrollo de sistemas de mediana envergadura a partir de propuestas teóricas, adquieren experiencia en el contacto con el material de investigación y en la discusión de las ideas, y contribuyen al enriquecimiento de la cátedra. En síntesis, la propuesta es brindar los conocimientos básicos, experiencia, y las herramientas necesarias para continuar con un autoaprendizaje.

Los contenidos del curso no difieren, en su introducción, del contenido tradicional. Se agrega como tema de importancia el modelado con curvas y superficies algebraicas, esencialmente los métodos de Bézier [13] y β -Splines [4]. Este tema insume un porcentaje considerable del tiempo de cursado, y los trabajos prácticos son bastante extensos, pero consideramos que es uno de los temas de aplicación más importantes en la computación gráfica, y que no puede estar ausente en un curso introductorio. Uno de los temas más difíciles en computación gráfica, pero que es esencial para ganar realismo sin perder interactividad, es el de cara oculta. En el Grupo se ha desarrollado un algoritmo que resuelve adecuadamente el problema para una clase lo suficientemente amplia de casos, el cual puede implementarse a lo largo del curso [11]. Otro tema que es profundizado es el de los modelos de iluminación y sombreado, incluyéndose una introducción al *ray tracing* y otros modelos no locales. Se finaliza con una exposición breve, enfocada hacia el interés de los alumnos, sobre sistemas de animación y métodos procedimentales (fractales, sistemas de partículas, modelos gramáticos):

- Introducción a los sistemas de *hardware* y *software* gráficos de una computadora.
- Desarrollo de programas gráficos de base (líneas y círculos).
- Geometría básica en 2D. Transformaciones.
- Espacios de objeto y de imagen. *Clipping*, *windowing* y *viewport*.
- Geometría en 3D. Transformaciones. Perspectiva.
- Línea y cara oculta.
- Modelos de iluminación y sombreado.
- Modelado con curvas y superficies algebraicas (Bézier, B-Splines y β -Splines).
- Modelos de iluminación no locales (*ray-tracing*, radiosidad).
- Introducción a los modelos procedimentales.
- Introducción a los sistemas de animación.

Con respecto a los trabajos prácticos, se enfatiza su desarrollo en comisiones de dos personas, de modo de poder distribuir la carga de la implementación pero no por ello

desconocerla por completo. Se atiende fundamentalmente el aspecto creativo en los trabajos prácticos, no prohibiéndose el intercambio de módulos entre las comisiones. Como los programas de los primeros prácticos constituyen la base de los prácticos siguientes, los alumnos aprenden por experiencia las bases y dificultades de la ingeniería de *software*, sobre todo en lo concerniente al reuso y portabilidad de sistemas.

El requisito de lectura y exposición de un *paper* 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 a lo largo de su carrera. El proceso de selección comienza con la elección de un tema por parte de la comisión. En función del grado de dificultad del tema elegido, y del concepto que la comisión exhibió durante el cursado, la cátedra elige un trabajo de investigación adecuado. La promoción del curso consiste en implementar las ideas de dicho trabajo en un nivel de profundidad acorde con la dificultad del tema.

Un curso con las características mencionadas se viene ofreciendo desde 1986 en la UNS como optativa en Ingeniería Electrónica, con un porcentaje bastante alto de elección por parte de los alumnos. Lo mismo sucede desde 1993 en la Licenciatura en Ciencias de la Computación. La circunstancia por la cual se ha mantenido un relativo éxito en el cumplimiento de los objetivos básicos y al mismo tiempo en satisfacer las expectativas cada vez más exigentes de los alumnos, no se debe a la elección de una currícula adecuada, sino a que la misma es deliberadamente abierta a las inquietudes.

5 Cursos optativos

Si bien el contenido del curso introductorio presentado puede ser ambicioso, muchas áreas importantes de la computación gráfica quedaron fuera del mismo. Para aquellos interesados en completar su formación, se diseñaron dos cursos optativos. El primero está destinado a la presentación de los problemas de investigación teóricos y aplicados de mayor relevancia en computación gráfica. Por dicha razón, el nombre del curso es *Tópicos Avanzados en Computación Gráfica*. Este curso fue dictado por primera vez en 1994 para un grupo reducido de alumnos. Un programa sintético del mismo es:

- **Temas Avanzados en Aproximación de Superficies.** Modelado de superficies de forma libre con sistemas de coordenadas auxiliares [3, 36]. Generalización de las superficies de Bézier para el modelado de topologías arbitrarias [23, 6].
- **Modelos de Iluminación y Sombreado.** Mapas de texturas, mapas de desplazamiento y mapas de normales [14, 41]. Espacios cromáticos, modelos de color, paletas estáticas, dinámicas y adaptativas [14, 10, 9].
- **Ray-Tracing.** *Ray-tracing* como solución al problema general de iluminación y sombreado [42]. Métodos generales de aceleración del algoritmo de *ray-tracing* [1, 2, 16, 18].
- **Visualización Científica.** Uso del color para la representación de mapas univariados y multivariados. Utilización de espacios cromáticos perceptuales [34, 40, 33]. Representación de datos volumétricos con *ray casting* y algunas consideraciones

de procesamiento de información [12]. Visualización de objetos tridimensionales translúcidos y opacos con *V-Buffer* [20, 39].

- **Síntesis de Imágenes con Conjuntos Fractales.** Simulación de fenómenos naturales con fractales no determinísticos y algoritmos estocásticos [15, 25, 8]. Simulación de árboles, plantas y objetos vivos con fractales autosimilares, gramáticas, y sistemas de reescritura [37, 31, 5].
- **Animación por Computadora.** Técnicas básicas de animación convencional y su aplicabilidad en computación gráfica [24]. Animación con *key-frames* e interpolación con *in-betweening* automático [32, 22].

Un segundo curso optativo, orientado más a las aplicaciones que a los aspectos teóricos, se denomina **Introducción a la Visualización Científica**. La *visualización científica* representa la culminación de las actuales posibilidades de los sistemas de computación gráfica. 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. Estos datos pueden provenir de sensores, como en el caso de tomógrafos o de satélites, o bien pueden provenir de tareas computacionales anteriores, como por ejemplo de simulaciones o de análisis por elemento finito. El resultado gráfico que se espera de la visualización de estos datos no es meramente *cuantitativo* –no se busca necesariamente la representación fiel de valores– sino *cualitativo* –se busca un entendimiento global de determinadas propiedades de los datos–. Estos objetivos son sumamente exigentes en términos de tecnología, tanto de hardware como de software. Un resumen adecuado del impacto de la visualización en distintos ámbitos de trabajo y áreas de conocimiento puede encontrarse reseñado en [7] y [35]. Este curso se dictará por primera vez en 1995.

Un alumno de la Licenciatura o de una Ingeniería que haya tomado el curso introductorio y por lo menos uno de los cursos avanzados estará en óptimas condiciones de encarar un adecuado trabajo final de grado en computación gráfica. Entre otros temas, podemos citar aquellos que en la actualidad pueden ser los más importantes para un trabajo final:

- Aproximación de superficies de topología arbitraria.
- Modelos híbridos y no locales de iluminación (*ray tracing*, radiosidad, ecuación del *rendering*).
- *Rendering* de volúmenes.
- Sistemas de animación.
- Modelos procedimentales (fractales, funciones no lineales, mapeos procedimentales).

6 Conclusiones

En este trabajo se presentaron alternativas para la enseñanza introductoria y avanzada en computación gráfica. Con respecto a un curso introductorio, se propone un equilibrio

entre la currícula tradicional y las propuestas modernas, tratando de balancear las ventajas y desventajas observadas en cada una. Con respecto a la enseñanza avanzada se presentan los contenidos de dos cursos optativos que posibilitan una mejor formación y acercamiento a los temas de investigación.

Referencias

- [1] T. Akimoto, K. Mase, y Y. Suenaga. Pixel-Selected Ray Tracing. *IEEE Computer Graphics and Applications*, 11(4):14–22, 1991.
- [2] J. Arvo y D. Kirk. Fast Ray Tracing by Ray Classification. *ACM Computer Graphics*, 21(4):55–64, 1987.
- [3] A. H. Barr. Global and Local Deformation of Solid Primitives. *ACM Computer Graphics*, 18(3):21–30, 1984.
- [4] R. Bartels, J. Beatty, y B. Barsky. *An Introduction to Splines for Use in Computer Graphics and Geometric Modelling*. Springer-Verlag, New York, 1987.
- [5] S. Casey y N. Reingold. Self-Similar Fractal Sets: Theory and Procedure. *IEEE Computer Graphics and Applications*, 14(3):73–82, 1994.
- [6] T. de Rose y R. Goldman. Functional Composition Algorithms via Bloosoming. *ACM Transactions on Graphics*, 12(2):115–135, 1993.
- [7] T. A. Defanti, M. D. Brown, y B. H. McCormick. Visualization: Expanding Scientific and Engineering Research Opportunities. En G. M. Nielson y B. D. Shriver, editores, *Visualization in Scientific Computing*, páginas 32–47, IEEE Computer Society Press, Los Alamitos, CA, 1990.
- [8] C. Delrieux, J. Belloso, y M. Marrochi. Síntesis de Fenómenos Naturales con Control Local de la Dimensión Fractal. En *XXI JAIIO, Jornadas Argentinas de Informática e Investigación Operativa*, páginas 43–62, SADIO, Buenos Aires, 1992.
- [9] C. Delrieux, M. Pascualetti, y M. E. Llorente. *Esquemas Estáticos de Color*. Reporte Técnico UNSIP-93-03, UNS, Departamento de Ingeniería Eléctrica, 1993.
- [10] Claudio Delrieux. *El Color en Computación Gráfica*. Reporte Técnico UNSIP-91-01, UNS, Departamento de Ingeniería Eléctrica, 1991.
- [11] Claudio Delrieux, Daniel Formica, Fernando Caba, y Esteban Pedroncini. Una Solución Eficiente al Problema de la Cara Oculta. *Revista Telegráfica Electrónica*, 934, 1991.
- [12] R. Drebin, L. Carpenter, y P. Hanrahan. Volume Rendering. *ACM Computer Graphics*, 21(4):65–74, 1988.
- [13] Gerald Farin. *Curves and Surfaces for Computer Aided Geometric Design*. Academic Press, New York, 1988.

- [14] J. Foley y A. Van Dam. *Fundamentals of Interactive Computer Graphics*. Addison-Wesley, Reading, Massachusetts, segunda edición, 1987.
- [15] A. Fournier, D. Fussell, y L. Carpenter. Computer Rendering of Stochastic Models. *Communications of the ACM*, 25(6):371–384, 1982.
- [16] A. Fujimoto, T. Tanaka, y K. Iwata. ARTS: Accelerated Ray Tracing System. *IEEE Computer Graphics and Applications*, 6(4):16–26, 1986.
- [17] W. K. Giloi. *Interactive Computer Graphics - Data Structures, Algorithms, Languages*. Prentice-Hall, Englewood Cliffs, New Jersey, 1978.
- [18] A. S. Glassner. Space Subdivision for Fast Ray Tracing. *IEEE Computer Graphics and Applications*, 4(10):15–22, 1984.
- [19] Scott Grissom, Jack Bressenham, Bill Kubitz, G. Scott Owen, y Dino Schweitzer. Approaches to Teaching Computer Graphics. En *SIGCSE '95 Proceedings*, páginas 382–383, ACM SIGCSE, ACM Press, Nashville, TN, 1995.
- [20] J. Kajiya y B. Von Herzen. Ray Tracing Volume Densities. *ACM Computer Graphics*, 18(4):91–102, 1984.
- [21] Peter Keller y Mary Keller. *Visual Cues: Practical Data Visualization*. IEEE Computer Society Press, Los Vaqueritos, CA, 1990.
- [22] J. Lasseter. Principles of Traditional Animation Applied to 3D Computer Animation. *ACM Computer Graphics*, 13(4):35–4, 1987.
- [23] C. T. Loop y T. de Rose. Generalized B-Spline Surfaces of Arbitrary Topologies. *ACM Computer Graphics*, 24(4):347–356, 1990.
- [24] N. Magnenat-Thalmann y D. Thalmann. *Computer Animation: Theory and Practice*. Springer-Verlag, Tokyo, segunda edición, 1988.
- [25] G. S. Miller. The Definition and Rendering of Terrain Maps. *ACM Computer Graphics*, 20(4):39–48, 1986.
- [26] W. Newman y R. Sproull. *Principles of Interactive Computer Graphics*. McGraw-Hill, New York, 1973.
- [27] G. M. Nielson y B. D. Shriver. *Visualization in Scientific Computing*. IEEE Computer Society Press, Los Alamitos, CA, 1990.
- [28] G. Scott Owen. A Workshop on Computer Graphics for Undergraduate Faculty. *Computer Graphics*, 25(3):179–185, 1991.
- [29] G. Scott Owen. ACM SIGGRAPH Education Committee Activities for Computer Graphics Educators. *Computer Graphics*, 28(3):179–182, 1994.
- [30] G. Scott Owen, Maria Larrondo-Petrie, y Cary Laxer. Computer Graphics Curriculum: Time for a Change? *Computer Graphics*, 28(3):183–185, 1994.
- [31] P. Prusinkiewicz, A. Lindenmayer, y J. Hanan. Developmental Models of Herbaceous Plants for Computer Imaginary Purposes. *ACM Computer Graphics*, 22(3):141–150, 1988.

- [32] W. Reeves. Inbetweening for Computer Animation using Moving Point Constraints. *ACM Computer Graphics*, 15(3):263–26, 1981.
- [33] P. K. Robertson. Visualizing Color Gamuts: a User Interface for the Effective Use of Perceptual Color Spaces in Data Displays. *IEEE Computer Graphics and Applications*, 8(5):50–64, 1988.
- [34] P. K. Robertson y J. O’Callaghan. The Generation of Color Sequences for Univariate and Bivariate Mapping. *IEEE Computer Graphics and Applications*, 6(2), 1986.
- [35] L. Rosenblum. Scientific Visualization at Research Laboratories. *IEEE Computer*, 22(8):68–100, 1989.
- [36] T. W. Sedberg y S. R. Parry. Free-Form Deformation of Solid Geometric Models. *ACM Computer Graphics*, 20(4):151–160, 1986.
- [37] A. J. Smith. Plants, Fractals and Formal Languages. *ACM Computer Graphics*, 18(3):1–10, 1984.
- [38] Karen Sullivan. A Second Generation Computer Graphics Course for Mathematics and Computer Science. *Computer Graphics*, 28(3), 1994.
- [39] C. Upson y M. Keeler. V-Buffer: Visible Volume Rendering. *ACM Computer Graphics*, 22(4):59–64, 1988.
- [40] C. Ware. Color Sequences for Univariate Maps: Theory, Experiments and Principles. *IEEE Computer Graphics and Applications*, 8(5):41–49, 1988.
- [41] Alan Watt y Mark Watt. *Advanced Animation and Rendering Techniques*. Addison-Wesley, London, 1992.
- [42] T. Whitted. An Improved Illumination Model for Shaded Displays. *Communications of the ACM*, 23(6):343–349, 1980.
- [43] Zhigang Xiang. A Nontraditional Computer Graphics Course for Computer Science Students. *Computer Graphics*, 28(3):186–188, 1994.