

Holografía Generada por Computadora: Un Nuevo Procedimiento

G. A. Patow^{*}, E. Aguirre[†], L.V. Bourimborde[‡], L. M. Zerbino[§] y M. Garavaglia^{**}
Centro de Investigaciones Ópticas (CIOp),
C.C. 124, Correo Central, 1900, La Plata, Argentina.

Resumen:

Se presenta un nuevo modelo para el cálculo de hologramas sintetizados computacionalmente. Dicho modelo se basa en la representación exacta del proceso de generación físico de un holograma para un número arbitrario fuentes genéricas (Ondas planas, esféricas o cualquier frente de onda del cual pueda extraerse la información de amplitud y fase en forma local, i.e.: como una $f(\mathbf{x})$, con \mathbf{x} un vector del espacio \mathcal{R}^3 , y el dominio de f en \mathcal{C} , los números complejos). Una vez sintetizado, será proyectado sobre una pantalla mediante un proyector de lámina líquida, fotografiado y su diapositiva utilizada como holograma a través del cual pasa un haz coherente de luz (generada con un láser). Los resultados son extremadamente satisfactorios, reproduciéndose físicamente las fuentes virtuales que dieron origen a los campos simulados.

Palabras Clave: Holografía, Holografía Sintetizada Por Computadora, Interferogramas, Computación Gráfica, Programación Orientada a Objetos.

^{*} También en LIFIA, Dto. Informática, UNLP, CC 11, La Plata, Argentina. Este trabajo fué realizado gracias a un subsidio de la Fundación Antorchas.

[†] Profesor en la Universidad Tecnológica Nacional, Villa Domínico, Argentina.

[‡] UNLLe Investigador del CONICET, Argentina.

[§] Profesor en UNLP e Investigador del CONICET.

^{**} Profesor en UNLP e Investigador Principal del CONICET.

Trabajo parcialmente financiado por PID-CONICET 3071800/91/92 y PID-BID-CONICET 1116/91.
CIOp pertenece al Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET) y Comisión de Investigaciones Científicas de la Provincia de Buenos Aires (CIC), Argentina.

1 Introducción.

La holografía desencadenó una intensa investigación desde su aparición práctica en 1960 y hasta principios de los setentas¹, introducida por los tres trabajos clásicos de Denis Gabor publicados hacia el final de la década del cuarenta.

Un holograma puede describirse básicamente como un interferograma codificado de forma tal que puede obtenerse de él una imagen a tiempo posterior. En esencia, esto significa que un holograma es construido de modo que la información de fase asociada al patrón de difracción del objeto es guardada durante la etapa de grabación del proceso interferencial. Por lo tanto, un holograma es un registro de la distribución de intensidades que describe la interferencia espacial entre el campo del objeto (un patrón de difracción de Fresnel o de Fraunhofer que surge de la transmisión o reflexión desde el objeto) y un campo de referencia (ondas planas o esféricas, tanto coaxiales cuanto formando un ángulo con el eje óptico del sistema)².

Debido a su ductilidad y capacidad de formar imágenes con información completa acerca de las fuentes (intensidad y fase, a diferencia de la fotografía tradicional que sólo almacena intensidades), los hologramas se han convertido en poderosas herramientas para conformación de haces mediante elementos ópticos holográficos, interferometría, computación óptica y una variedad de aplicaciones diversas. Precisamente por su papel como moduladores espaciales del frente de onda, la posibilidad de generarlos mediante el uso de computadoras ha cobrado un interés enorme³.

Los intentos de generación de hologramas por computadora se remontan prácticamente a los mismos principios de la holografía (principios de los setentas), basados principalmente en el uso de técnicas de Fourier para lograr los hologramas de transformada de Fourier⁴ (ver sección 2 de este trabajo). Los mismos se basaban en utilizar un monitor de vídeo para mostrar los hologramas generados, generalmente de muy baja resolución (hasta 128×128 muestras, dada las limitaciones de hardware de la época). Estos eran fotografiados a una distancia tal que el diagrama de Fourier sólo ocupase una pequeña porción de la superficie de la película sensible de muy alta resolución (típicamente 1000 líneas por milímetro). Luego de ser revelado el material fotográfico, el área de la diapositiva cubierto por el diagrama era usado como holograma de transmisión, haciendo pasar a través de él un haz coherente cuasimonocromático⁵. Los posibles valores utilizados (escala de grises) en general era solamente de dos (blanco y negro), aunque Lee, en su trabajo original, sugiere la utilización de la gama completa ofrecida por el monitor⁶.

Más adelante, para superar las limitaciones provocadas por la baja resolución del monitor (en píxeles), los hologramas comenzaron a generarse en forma binaria (sólo dos valores posibles, en general blanco y negro), graficándose en hojas de papel de gran tamaño, de las que se tomaban las correspondientes fotografías. Aún así, el proceso de fotografiado, revelado y reconstrucción continuó siendo el mismo. Los algoritmos generalmente computaban el valor del holograma (generalmente de transformada de Fourier) en la superficie de la película, utilizando una red de celdas idéntica a un arreglo de píxeles en un display de monitor. Dentro de cada celda, se dibujaban sistemas de rectángulos negros sobre fondo blanco. La disposición de dichos rectángulos dependía de la fase del campo en el punto, mientras que su tamaño estaba relacionado con la intensidad (recordar que el campo lumínico en un punto está descrito por un número complejo). Esta técnica aún continúa utilizándose hoy en día para generar hologramas artificiales por computadora^{7,8,9}. Otras posibilidades incluyen la utilización de equipo especializado, como moduladores de luz espaciales¹⁰, microlitografía¹¹, etc.

En trabajos recientes¹², se expusieron la caracterización de un proyector de TV de lámina líquida y su utilización como reemplazo de los aspectos mecánicos (graficado) del proceso de generación de hologramas. Dicho proyector (llamado eidóforo, del griego "portador de imágenes") se basa en la utilización de una lámina de aceite como soporte de una red de difracción sinusoidal escrita por tres haces de electrones portadores del código RGB¹³. Dicha señal puede recibirse como señal de televisión, o, mediante un codificador adecuado, ser controlada desde una computadora personal. Así puede controlarse completamente la señal del proyector, pudiendo utilizárselo para proyectar hologramas artificialmente generados. En este trabajo describiremos los aspectos computacionales desarrollados para generar hologramas por computadora, aprovechando todo el potencial de un proyector multinivel (8 bits, pero el proyector limita el rango a 6 bits, es decir, 64 niveles diferentes).

En la siguiente sección describiremos los principios básicos involucrados en la generación de hologramas, y en la Sección 3 veremos cómo éstos principios se aplican para lograr un nuevo modelo para la generación de Hologramas sintéticos. En la Sección 4 presentaremos los detalles de la implementación de dicho sistema, que hemos denominado **HOLOG**. En la Sección 5 se presentará una breve justificación del modelo elegido en contraste con otros modelos de uso frecuente en Computación Gráfica. Finalmente serán discutidas las conclusiones.

2 Principios básicos de la Holografía.

Como mencionásemos más arriba, un holograma básicamente puede describirse como un interferograma, construido de forma tal que la información de fase asociada al patrón de difracción del objeto es guardada durante la etapa de grabación¹⁴. De esta forma, un holograma es un registro de la distribución de intensidades que describe la interferencia espacial^{††} entre el campo del objeto $\Psi_d(x)$ (un patrón de difracción de Fresnel o de Fraunhofer que surge de la transmisión o reflexión desde el objeto) y un campo de referencia $\Psi_r(x)$ (ondas planas o esféricas, tanto coaxiales cuanto formando un ángulo con el eje óptico del sistema), luego:

$$\begin{aligned} \text{Holograma} \rightarrow I(x) &= |\Psi_d(x) + \Psi_r(x)|^2 & (1) \\ &= |\Psi_d(x)|^2 + |\Psi_r(x)|^2 + \Psi_d(x) \Psi_r^*(x) + \Psi_d^*(x) \Psi_r(x) \end{aligned}$$

Los hologramas suelen ser clasificados en varias formas y desde varios puntos de vista. Aquí seguiremos la clasificación dada por Reynolds et al, considerando el proceso de formación físico:

1. Si el plano del holograma se encuentra en la región de difracción de Fresnel del objeto, y el haz de referencia es coaxial con el campo de iluminación, el holograma resultante se denomina *Holograma de Fresnel*.
2. Si se encuentra en la región de Fraunhofer, y el haz de referencia es coaxial con campo de iluminación, se denomina *Holograma de Fraunhofer*.
3. Si el holograma está en la región de Fresnel (Fraunhofer), pero el haz de referencia forma un ángulo con el eje óptico, se denomina *Holograma lateral de Fresnel (Fraunhofer)*.

^{††} El patrón de interferencia es espacial porque a las frecuencias ópticas, todas las variaciones temporales asociadas con los campos se promedian a cero por el receptor (película fotográfica).

4. Si la intensidad dada por la Ec. (1) es de la forma $I(x) = |K + \underline{D}(x)|^2$, donde $\underline{D}(x)$ denota la transformada exacta de Fourier del objeto, el holograma resultante se denomina *Holograma de Transformada de Fourier*.
5. Si el haz de referencia fuera del eje se mezcla con una imagen coherente, el holograma se denomina *Holograma de Imagen*.

Para reconstruir la imagen del objeto congelada en un holograma, se expone el holograma a un haz óptico coherente, formándose entonces una imagen en una posición que dependerá del tipo de holograma de que se trate. Durante la reconstrucción, los dos primeros términos de la Ec. (1) generalmente describen una imagen irreconocible de distribuciones centrada en el eje óptico, mientras que los términos con productos cruzados describen imágenes conjugadas (consideraremos tanto reales cuanto virtuales) los que son fácilmente reconocibles.

Para describir más detalladamente las propiedades de un holograma, describiremos ahora el holograma producido por un objeto formado por dos fuentes puntuales coherentes de intensidad $a \cdot \delta(\xi, z_1)$ y $a \cdot \delta(\xi, z_2)$ ubicadas a distancias z_1 y z_2 frente al plano del holograma, como se muestra en la Figura 1 (ξ es la frecuencia de las fuentes). Se supone un haz de referencia fuera del eje que es coherente con las fuentes.

La distribución de intensidades a lo largo del eje x en el plano del holograma, $I(x)$, grabada en la película fotográfica, estará dada (de acuerdo con la Ec. (1)), por:

$$\begin{aligned}
 I(x) &= |K \exp(ik\theta_0 x) + A a \exp(ikx^2/2z_1) + A a \exp(ikx^2/2z_2)|^2 \\
 &\approx K^2 + A^* K a^* \exp(ik\theta_0 x) [\exp(-ikx^2/2z_1) + \exp(-ikx^2/2z_2)] \\
 &\quad + A K a^* \exp(-ik\theta_0 x) [\exp(ikx^2/2z_1) + \exp(ikx^2/2z_2)]
 \end{aligned} \tag{2}$$

en esta última expresión, A incluye los factores de oblicuidad de Fresnel en la integral de difracción de Rayleigh-Sommerfeld, K es la amplitud de la onda de referencia, y $K \gg a$ de forma tal que el término correspondiente a a^2 es despreciable. Esto, a su vez, asegura que la impresión de la película fotográfica será en su región lineal. La Ecuación (2) implica que cada fuente puntual da origen a una onda esférica de diferente radio de curvatura en el plano de difracción.

Revelar la película fotográfica y ubicarla en el sistema de reconstrucción esquematizado en la Figura 2, nos permite usar la Ec. (2) como condición de contorno para la subsiguiente integral de difracción de Rayleigh-Sommerfeld, la que es empleada para determinar la distribución de amplitudes a una distancia z_3 del plano del holograma sobre el plano α .

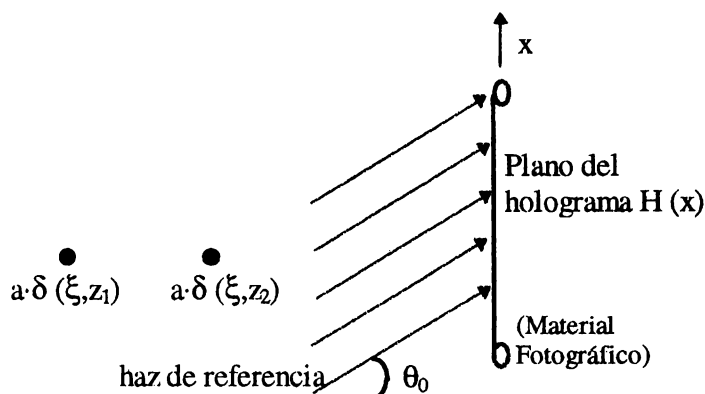


Figura 1: Esquema del grabado de un holograma de dos fuentes puntuales longitudinales separadas.

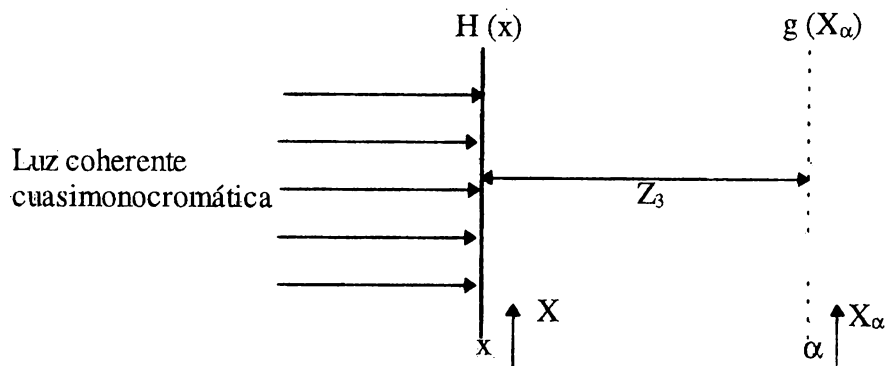


Figura 2: Sistema de reconstrucción

Consideremos primeramente la contribución del segundo término en la Ec. (2) al resultado final. Ignorando constantes multiplicativas (sólo nos interesa una descripción cualitativa del fenómeno, en aras de su mejor entendimiento), la amplitud reconstruida a lo largo del eje X_α está dada por la expresión:

$$g(X_\alpha) = \int \left\{ \exp(ik\theta_0 x) \left[\exp(-ikx^2/2z_1) + \exp(-ikx^2/2z_2) \right] \exp(ik(x-X_\alpha)^2/2z_3) \right\} dx$$

$$= \exp(ikX_\alpha^2/2z_3) \int \left[\exp(-ikx^2/2z_1) + \exp(-ikx^2/2z_2) \right] \exp(ikx^2/2z_3) \times \exp(-ikx(X_\alpha/z_3 - \theta_0)) dx$$

En la posición $z_3 = z_1$, el primer término de la integral nos da $\delta(\alpha - \theta_0 z_1)$, y en la posición $z_3 = z_2$, el segundo nos da $\delta(\alpha - \theta_0 z_2)$. Por lo tanto, ambas fuentes puntuales son reconstruidas en sus correspondientes puntos de imagen, a la misma distancia del holograma a la que estaban durante el proceso de grabado. Las imágenes están fuera del eje a una distancia $\theta_0 z_1$ y $\theta_0 z_2$ respectivamente, de forma tal que están alineadas una atrás de la otra en la dirección θ_0 . Este ejemplo nos mostró claramente que el almacenamiento de la información de fase del objeto durante la grabación retiene la naturaleza tridimensional de la imagen. Por ello, ondas esféricas diferentes en el plano del holograma implican que sus respectivas fuentes puntuales estaban localizadas a diferentes distancias del holograma.

La reconstrucción del tercer término en la Ec. (2) conduce a dos imágenes puntuales, $\delta(X_\alpha + \theta_0 z_1)$ y $\delta(X_\alpha + \theta_0 z_2)$, ubicadas a distancias $z_3 = -z_1$ y $z_3 = -z_2$, es decir, imágenes virtuales detrás del holograma (ésta es la imagen que se observa normalmente en un holograma porque el ojo enfoca más fácilmente la imagen virtual).

El procedimiento hasta aquí descrito puede extenderse para un arreglo continuo de fuentes puntuales simulando la luz reflejada por un objeto tridimensional. Por ejemplo, si expresamos al objeto tridimensional como

$$O(\xi, z) = \sum_{n=1}^{\infty} A_n \delta(\xi_n, z_n) \quad , \quad (3)$$

por lo que la distribución de intensidades en el holograma será

```

Pixel.GetPos(Screen): Vector3D;

RealX ← x / Screen.GetResX - 0.5; {en [-0.5 .. 0.5]}
RealY ← y / Screen.GetResY - 0.5; {en [-0.5 .. 0.5]}

RealX ← RealX * Screen.GetDimX;
RealY ← RealY * Screen.GetDimY;

Return ( [RealX, RealY, 0, 1] );
                {en z=0, plano xy}

```

Figura 3: Expresiones para el mapeo
Pantalla → Película Fotográfica

$$I(x) = |K \exp(ik\alpha_0 x) + A \sum_{n=1}^{\infty} A_n \delta(\xi_n, z_n)|^2 \quad (4)$$

con K y A dos constantes adecuadas. Puede apreciarse fácilmente como todas estas expresiones se apoyan, implícitamente, en la linealidad de las ecuaciones de Maxwell, que nos permite usar el Principio de Superposición para sumar las amplitudes de las diferentes contribuciones.

3 Un nuevo modelo de Holografía Generada por Ordenador.

Aquí presentaremos brevemente nuestro nuevo modelo para generar hologramas por computadora, y en la sección subsiguiente desarrollaremos los detalles de su implementación. Hasta ahora, sólo presentamos la formulación física del problema, con lo que están dadas todas las herramientas necesarias para su desarrollo.

Como fue descrito en la introducción, los modelos existentes hasta el momento están basados en la fotografía de un dibujo de un patrón de difracción calculado por computadora sobre una hoja de papel de dimensiones considerables mediante sistemas de graficación monocromático y su posterior procesado físico químico. La diapositiva resultante actúa como un holograma de transmisión. Los hologramas así obtenidos estaban limitados al uso de sólo dos valores de intensidad: Blanco y Negro. Pero, si reemplazamos la etapa de graficado por la proyección de una imagen mediante un proyector de TV de lámina líquida, siendo dicha imagen posteriormente fotografiada, podemos utilizar la gamma completa de valores disponibles por el proyector. En nuestro caso, el proyector estaba conectado a una placa de vídeo cuyas imágenes eran generados por una computadora, lo que nos permitía disponer de un rango efectivo de 6 bits (64 tonos), que representa un incremento notable respecto de los dos valores anteriormente utilizados.

Para poder calcular un holograma, será necesario, en la práctica, encontrar alguna forma de resolver la Ecuación (4) para cada punto del plano del mismo. Dicho plano, por convención, estará ubicado sobre el plano xy en $z = 0$, en un espacio euclídeo \mathcal{R}^3 , estando los entes geométricos que actúan como fuentes de luz en cualquier lugar de dicho espacio. Siguiendo una convención ampliamente aceptada en Computación Gráfica¹⁵, sólo consideraremos como válidas aquellas fuentes (entes geométricos) que tengan $z < 0$. De esta forma, simularemos más acertadamente una película fotográfica real, dónde sólo hay material fotosensible en un lado de la misma.

Para cada pixel P de la pantalla Para cada fuente de luz j $\Psi_{total}(P) \leftarrow \Psi_{total}(P) + \Psi_j(P)$ $Valor(P) \leftarrow \Psi_{referencia}(P) + \Psi_{total}(P) ^2$

Figura 4: Pseudocódigo de un sistema holográfico genérico.

Una vez definidas estas convenciones, es necesario realizar un mapeo entre la película fotográfica real (que, a fines prácticos, podemos considerar como continua) y su realización física: la pantalla (discretizada en píxel). Dicho mapeo se realiza mediante las expresiones detalladas en la Figura 3, donde se presenta el pseudocódigo de GetPos, que es un mensaje entendido por cada píxel y que devuelve, en función de la resolución y dimensiones físicas de la pantalla/película fotográfica, el vector 3D correspondiente a su centro. Como puede verse, el vector devuelto es de la forma [a,b,c,1], debido a que, para su uso posterior, hemos decidido utilizar coordenadas homogéneas¹⁶ para representar a los vectores 3D.

Armados con esta operación, sólo resta calcular la Ec. (4) para cada píxel de la pantalla, y asignarle el valor resultante, como puede verse claramente del pseudocódigo de la Figura 4.

Para finalizar, es necesario considerar un último punto concerniente a la imagen final (holograma) a ser fotografiada: la saturación. Al igual que cualquier película fotográfica, los monitores y el sistema de proyección poseen un rango de valores posibles (en el caso del proyector, como ya mencionamos, estaba limitado a 64 valores diferentes), provocando para cualquier valor fuera de este rango diferentes efectos de "aliasing", siendo el más común el del recorte, donde los valores por encima de cierto umbral son asociados al máximo valor y los que se encuentran por debajo son asociados al mínimo. Para solucionar esto se decidió utilizar un algoritmo en dos pasadas, la primera guardando el máximo valor requerido, y la segunda (opcional) corrige los valores del holograma, normalizándolos de forma tal que su máximo valor se corresponda con el máximo del rango del dispositivo utilizado.

4 El sistema HOLOG.

En esta sección describiremos los detalles de implementación del sistema de generación por computadora de hologramas artificiales, **HOLOG**. Dadas las características del problema a enfrentar y la versatilidad del Paradigma de Programación Orientado a Objetos (POO), fue ésta la metodología elegida para desarrollar nuestro sistema.

Como puede verse de la Ecuación (4), el resultado final de intensidad en un punto de la película (plano α) es simplemente el cuadrado de la suma de todas las contribuciones de cada fuente puntual más la onda de referencia. Por lo tanto, desde el punto de vista matemático, será necesario crear las clases nuevas **Vector3D** y **Complejo**, para poder representar al dominio y al codominio de las funciones $\Psi_n^{fuente}(x)$ (las funciones de amplitud de las n fuentes luminosas, como están descritas en la Ecuación (1)). Entre las funciones mínimas que debe proveer la primera de las clases mencionadas, está el producto escalar (producto interno), cambios de escala (producto por un escalar) y el módulo, mientras que la segunda, debe soportar módulo, adición y operaciones de escala.

Los diferentes tipos de imágenes o luces, heredan todos de la clase **Source**, lo que provee la funcionalidad básica: Ajuste de amplitud y número de onda k (definido como $2\pi / \lambda$, donde λ es la longitud de onda de la fuente estudiada), y cálculo de la amplitud del campo originado por la fuente en un punto del espacio dado (puede comprenderse la importancia que tiene el Principio de Superposición, implícitamente usado aquí para superponer las ondas provenientes de diferentes fuentes). Para facilitar la posterior reproducción del holograma, se utilizó como longitud de onda λ para todas las fuentes, el valor 632,8 nm, debido a que el láser He-Ne utilizado emitía luz roja en dicha longitud.

De esta clase descienden todos los diferentes tipos de luces requeridos. En nuestra implementación actual, **Spherical** y **Reference** son las primitivas utilizadas para representar fuentes puntuales y ondas planas (usadas como haz de referencia) respectivamente. Las clases se corresponden con las correspondientes expresiones físicas:

$$\Psi_{\text{Esférica}}(\mathbf{r}) = \exp(\pm i k r) / r$$

$$\Psi_{\text{Onda Plana}}(\mathbf{r}) = \exp(\pm i \mathbf{k} \cdot \mathbf{r})$$

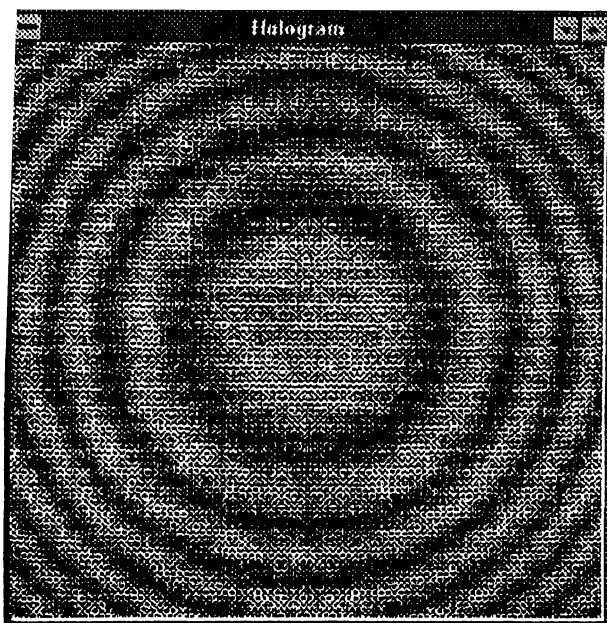
en estas expresiones, r es el módulo del vector \mathbf{r} , la distancia entre la fuente y el punto de observación ($\mathbf{r} \equiv \text{PosiciónObservacion} - \text{UbicaciónDeLaFuente}$) y \mathbf{k} es el vector número de onda, cuyo módulo es k y su dirección es la dirección de propagación de la onda plana. Obviamente, éstos son sólo dos tipos básicos utilizados en la óptica para modelar posibles fuentes, permitiendo la flexibilidad de la POO implementar cualquier tipo de fuente que se desee (e.g.: anisotrópicas, irregulares, etc.), con tal que pueda computarse de ella la amplitud como función de la coordenada del espacio donde se encuentra el punto de observación. Un ejemplo concreto de un tipo de fuente más sofisticada es la fuente-segmento: escribimos un segmento (definido por sus dos extremos, \mathbf{r}_1 y \mathbf{r}_2) en el espacio \mathcal{R}^3 como la combinación lineal $\mathbf{r}(t) = (1-t) \mathbf{r}_1 + t \mathbf{r}_2$, con t un parámetro perteneciente al intervalo $[0,1]$. Entonces, en virtud del Principio de Superposición, un segmento será la superposición de infinitas fuentes puntuales esféricas, es decir:

$$\Psi_{\text{Segmento}}(\mathbf{r}) = \int_0^1 \frac{\exp(\pm i k \|\mathbf{r}(t)\|)}{\|\mathbf{r}(t)\|} dt$$

En la implementación actual, este tipo de fuente ha sido resuelto como una sucesión de fuentes puntuales, reemplazando la integral por una sumatoria en la expresión anterior.

Para la entrada de los datos se decidió implementar un pequeño parser¹⁷ para un modesto lenguaje que permite especificar, mediante un archivo de texto, la resolución del dispositivo de salida (proyector), su rango, y la correspondencia entre éstos y los parámetros (tamaño) de la película fotográfica simulada, las diferentes fuentes utilizadas (un conjunto de fuentes tiene una única longitud de onda asociada) y, finalmente, el archivo se salida para su posterior reutilización.

Un pequeño ejemplo de archivo de entrada y el holograma resultante puede verse en la Figura 5: cuatro fuentes puntuales, formando una letra "Y", inclinada 45° con respecto al plano del holograma, y centradas a una distancia de 0.5 metros de la película (de 3×3 milímetros² de superficie). Como puede verse, la resolución se especifica en pixeles, el tamaño de la película en milímetros, las longitudes de onda se especifican en Angstroms (10^{-8} m) y las distancias en metros (o sea, las unidades más convenientes en cada caso).



```

Screen: 400*400pix, 4*4 mm, (64,64,64)grys,
ON:
Sources: 6328Angstroms {
  Spherical 0, -0.01, -0.5; 3
  Spherical 0, 0, -0.51; 3
  Spherical -0.01, 0.01,-0.52; 3
  Spherical 0.01, 0.01,-0.52; 3
  Reference 0,0,1; 5
}
SaveAs: "holo.bmp"

```

Figura 5: Holograma y archivo de entrada respectivo, generados mediante **HOLOG**. El archivo especifica cuatro fuentes puntuales (esféricas) y un haz de referencia. Las palabras "pix", "mm", "grys" y "Angstroms" son simplemente comentarios para el programador, e ignorados por el parser.

5 Justificación del modelo elegido.

En esta sección buscaremos dar una justificación al modelo elegido, comparándolo con otros modelos similares existentes en computación gráfica.

Una rápida mirada a la presentación del modelo hecha en la Sección 3 nos muestra la gran similitud que posee el modelo elegido con un algoritmo clásico de Trazado de Rayos¹⁸. Dicha similitud no es casual: Efectivamente, para computar la intensidad final de cada pixel, deben calcularse las contribuciones de todas las fuentes en la escena, así como las posibles oclusiones (ocultamientos por terceros objetos) y contribuciones indirectas (por reflexión/transmisión) que puedan ocurrir. Por ejemplo, el agregado del cálculo de sombras (interposición de un objeto entre la película y una fuente dada) se realiza agregando un par de líneas al código original (que realizan un testeo de intrsección rayo-objeto con todos los elementos que forman la escena), siempre que despreciemos los efectos de difracción en los bordes del objeto interpuesto. Dicha aproximación es viable en el sentido que estos efectos

```

Para cada pixel P de la pantalla
  Para cada fuente de luz j
    Si no hay objetos interpuestos entre P y j
       $\Psi_{total}(P) \leftarrow \Psi_{total}(P) + \Psi_j(P)$ 
  Valor(P)  $\leftarrow |\Psi_{referencia}(P) + \Psi_{total}(P)|^2$ 

```

Figura 6: Pseudocódigo de un sistema holográfico genérico incorporando oclusiones.

difractivos están circunscriptos, para las altas frecuencias usuales de trabajo en holografía, a una delgada franja que rodea a la sombra del objeto. Ver Figura 6.

De todas formas, uno podría verse tentado a intentar algún algoritmo de interpolación para acelerar los cálculos, en forma similar a como actúan los algoritmos Scanline con los modelos de iluminación de Gouraud o Phong¹⁹. Dicha interpolación no podría hacerse sobre la pantalla debido a que "borraría" las fluctuaciones que estamos interesados en calcular. Una posibilidad, entonces, sería interpolar los valores calculados, por ejemplo, en los extremos de una fuente en forma de segmento. Pero no debemos olvidar un importantísimo y crucial detalle: Al estar trabajando con luz cuasimonocromática, los efectos de interferencia se vuelven sumamente relevantes, al punto de afectar completamente nuestros cálculos.

En la Computación Gráfica tradicional, todas las fuentes de luz son incoherentes, por lo que, para calcular la intensidad resultante, debemos sumar intensidades en vez de amplitudes, y, en ese caso, en vez de la Ecuación (4) deberemos usar:

$$I(x) = |K|^2 + A^2 \sum_{n=1}^{\infty} A_n^2 |\delta(\xi_n, z_n)|^2$$

si utilizamos una fuente esférica, por ejemplo, su contribución será de la forma $I_{\text{Esférica}}(r) = A_n^2 / r^2$, que es la expresión ampliamente conocida para la contribución a la intensidad total por una fuente puntual (ver, por ejemplo, Watt y Watt). De la misma forma, una onda plana sólo contribuirá con una constante aditiva.

En cambio, para nuestro problema, las fases juegan un papel preponderante, obligándonos a tener mucho cuidado en su manejo por las fluctuaciones que generan al interferir con el haz de referencia y entre sí. Este hecho invalida prácticamente cualquier acercamiento similar a los algoritmos Scanline tradicionales que queramos hacer.

Por ello, un acercamiento al problema al estilo del Trazado de Rayos es, en comparación con la bibliografía existente, la única solución viable si se busca generar un holograma completo y funcional. Si sólo se busca una representación aproximada que nos permita mostrar las imágenes con ciertas restricciones, podemos intentar otro tipo de algoritmos. Un claro ejemplo ha sido desarrollado recientemente²⁰ en donde se sacrificó resolución y paralaje vertical en aras de la velocidad, con el fin de poder presentar estereogramas en tiempo real.

6 Discusión y conclusiones.

Un nuevo procedimiento para el cálculo y síntesis de Hologramas Generados por Computadora ha sido presentado. Dicho modelo se basa en una implementación que refleja la Física involucrada en el problema, junto con el empleo de un sistema de proyección de alta definición, que nos permite reemplazar los aspectos mecánicos por otros puramente óptico-digitales. Otro beneficio de este nuevo procedimiento es la posibilidad de generar hologramas con una mayor riqueza, puesto que el número de valores disponibles se ha ampliado considerablemente (en el sistema actual tenemos 64 valores contra los 2 originales). El problema estudiado obliga a realizar un cálculo detallado píxel por píxel, evitando cualquier tipo de interpolación que pudiese acelerar los cálculos debido a que se ignorarían los efectos interferenciales buscados.

Para lograr la mayor flexibilidad posible, así como también para facilitar la construcción de la simulación y el agregado de nuevos componentes, se ha optado por una codificación dentro del paradigma de la Programación Orientada a Objetos.

Los resultados experimentales son completamente satisfactorios, lográndose reproducir perfectamente en forma física los registros logrados. Ver Figuras 7 y 8.

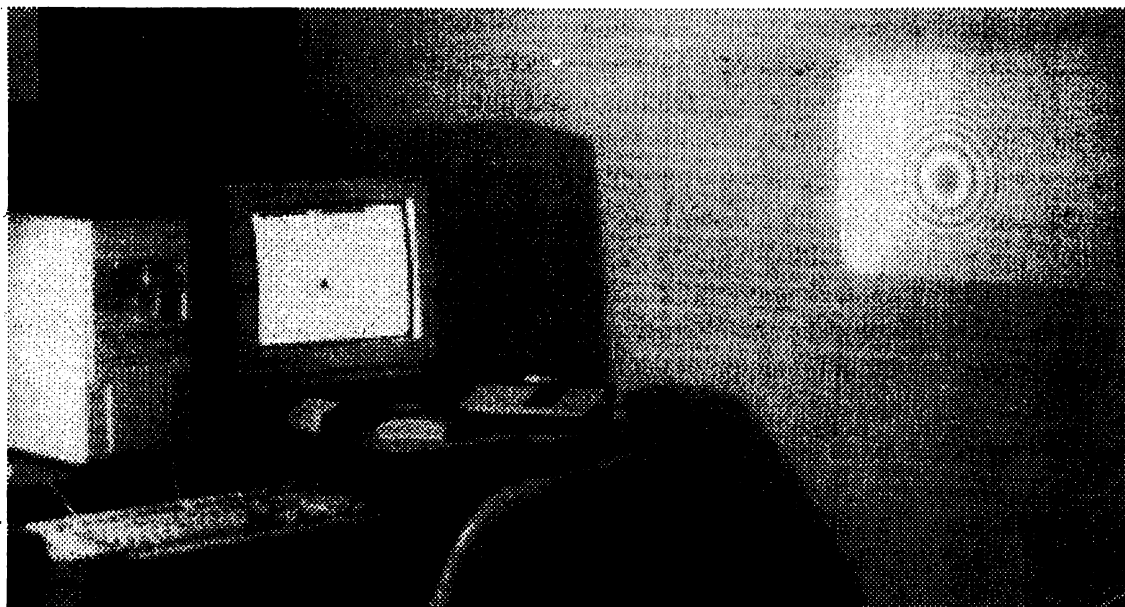


Figura 7: Instalación experimental: sobre la pantalla del monitor aparece el holograma generado por computadora de una fuente puntual y sobre la pantalla de proyección se visualiza el mismo holograma proyectado.

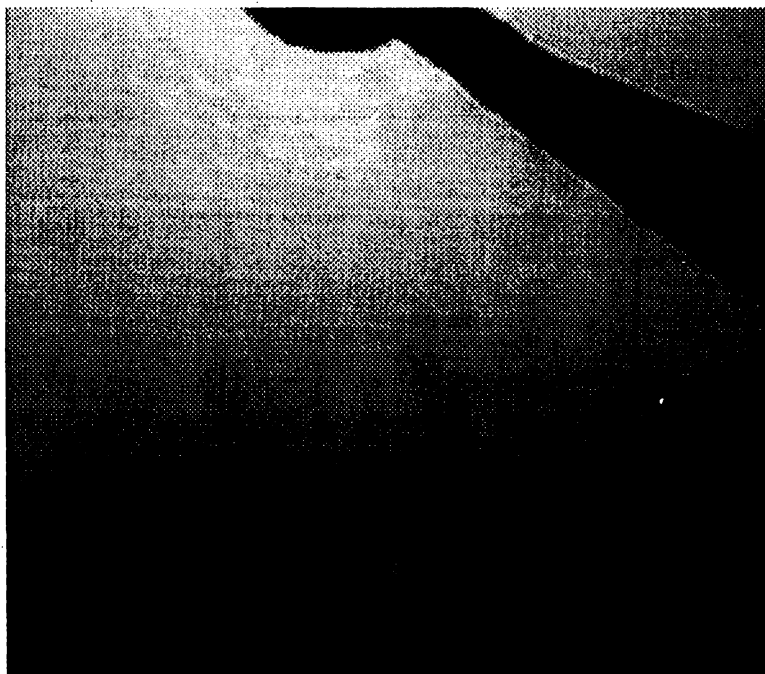


Figura 8: Reconstrucción del holograma de una fuente puntual ubicada en el espacio. Se bloquea el orden cero para evitar saturación de las películas. El punto brillante en la parte central-inferior de la imagen es la posición de los órdenes primeros correspondientes a las ondas convergentes.

Bibliografía.

- ¹ Hetch, E y Zajac, A "Optica", Fondo Educativo Interamericano, 1977.
- ² Françon, M. "Optique", Masson y Cie., Paris 1972.
- ³ Wyant, J, y Bennett, V. "Using Computer Generated Holograms to Test Aspheric Wavefronts", App. Opt., **11** (1972), pp. 2833; "Computer Generated Holograms for Testing Optical Elements", App. Opt., **10** (1971), pp. 619.
- ⁴ Tricoles, G. "Computer Generated Holograms: an historical review", App. Opt., **26** (1987), pp. 4351.
- ⁵ Lee, W. "Binary Synthetic Holograms", App. Opt., **13** (1974), pp. 1677.
- ⁶ Lee, W. "Sampled Fourier Transform Hologram Generated by Computer", App. Opt., **9** (1970), pp. 639.
- ⁷ Lohmann, A y Paris, D "Binary Fraunhofer Holograms, Generated by Computer", App. Opt. **6** (1967), pp. 1793.
- ⁸ Lohmann, A y Sinzinger, S. "Graphic Codes for Computer Holography", App. Opt., **34**(1995), pp. 3172.
- ⁹ Lohmann, A y Ojeda-Castañeda, J. "Computer generated holography: novel procedure", Opt. Comm., **103** (1993), pp. 181.
- ¹⁰ Rosen, J; Shiv, L, Stein, J y Shamir, J. "Electro-optic hologram generation on spatial light modulators" J. Opt. Soc. Am. A, **9** (1992), pp. 1159.
- ¹¹ Krackhardt, U; Streibl, N y Schwider, J. "Fabrication errors of computer generated multilevel phase-holograms". Optik, **95** (1994), pp. 137.
- ¹² Bourimborde, Patow, G; L; Zerbino, L; Aguirre, E, Laquidara y Gravaglia, M. "Caracterización del frente de onda de un modulador espacial de luz líquido-opto-electrónico bidimensional", Reunión de la Asociación de Física Argentina, Bariloche, Argentina, 1995.
- ¹³ Bourimborde, Patow, G; L; Zerbino, L; Aguirre, E, Laquidara y Gravaglia, M. "Empleo de un Modulador Líquido-Opto-Electrónico para Generar Diferentes Frentes de Onda", VIII Reunión Regional Latinoamericana de Astronomía, Montevideo, Uruguay, 1995.
- ¹⁴ Reynolds, George; DeVelis, John; Parrent Jr., George y Thompson, Brian. "The New Physical Optics Notebook: Tutorials in Fourier Optics", SPIE – American Institute of Physics, 1989.
- ¹⁵ Foley, J., A. van Dam, S. Feiner and J. Hughes, "Computer Graphics Principles and Practice, 2nd. Edition", Addison-Wesley, 1990.
- ¹⁶ Rogers, D. y Adams, J. "Mathematical Elements for Computer Graphics, 2nd. Edition", McGraw-Hill, NY, 1990.
- ¹⁷ Aho, A; Sethi, R y Ullman, J. "Compilers: Principles, techniques and tools", Addison-wesley, Reading, Massachusetts, 1986.
- ¹⁸ Glassner, A., "An Introduction to Ray Tracing", Academic Press, 1991.
- ¹⁹ Watt, A. and M. Watt, "Advanced Animation and Rendering Techniques. Theory and Practice", Addison-Wesley, 1992.
- ²⁰ Lucente, M y Galyean, T. "Rendering Interactive Holographic Images", Proceedings of SIGGRAPH'95, (Agosto 1994), 387-394.