

# Visualización autoestereoscópica de múltiples campos de luz

Miguel Escrivá, Joan Blasco, Francisco Abad, Emilio Camahort, Roberto Vivó

Departamento de Sistemas Informáticos y Computación  
Universidad Politécnica de Valencia  
Camino de Vera s/n  
46022 Valencia, Spain  
{mescriva, jblasco, fjabad, camahort, rvivo}@dsic.upv.es

---

## Abstract

*Los campos de luz se introdujeron hace una década como un nuevo modelo de representación gráfica de alta dimensión. Sin embargo, apenas se han utilizado debido a lo específico de sus aplicaciones y a su alto coste de almacenamiento. Recientemente se han relacionado los dispositivos espaciales con el campo de luz. Estos dispositivos permiten a múltiples usuarios ver imágenes 3D sin utilizar gafas u otros elementos intrusivos. En este artículo presentamos un modelo del campo de luz que se puede visualizar en un dispositivo espacial de tipo autoestereoscópico. El modelo es independiente del punto de vista y soporta multirresolución continua, visualización foveal e integración de múltiples campos de luz y geometría en una sola imagen. También demostramos que es posible manipular interactivamente una escena compuesta por varios campos de luz y modelos geométricos. En estos casos la visibilidad se resuelve correctamente. Nuestro objetivo es aplicar el modelo presentado a la televisión 3D y a la visualización espacial.*

Categories and Subject Descriptors (according to ACM CCS):

I.3.3 [Computer Graphics]: Picture/Image Generation–Display algorithms

I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling–Curve, surface, solid, and object representations

I.3.6 [Computer Graphics]: Methodology and Techniques–Graphics data structures and data types

---

## 1. Introducción

Los campos de luz son una técnica de modelado basada en imágenes. Almacenan la radiancia que emana de una escena sintética o real, muestreada a lo largo de un conjunto 4D de líneas orientadas. Los campos de luz pueden además ser extendidos con información de profundidad o *proxies* geométricos para reconstruir más fielmente la función continua de radiancia durante la visualización.

Los campos de luz tienen muchas ventajas. Permiten modelar objetos complejos capturados del mundo real sin necesidad de reconstruir su geometría. Se caracterizan por un tiempo de visualización que sólo depende de la complejidad de la imagen final y no de la geometría de la escena. Soportan modelos híbridos basados en geometría e imágenes, como se demuestra en este artículo. También permiten visualizar interactivamente modelos cuya visualización interactiva no es posible sobre una GPU moderna debido bien

a su complejidad geométrica, o bien a que se desea simular efectos de iluminación avanzados, que no se pueden conseguir en tiempo real. Finalmente, los campos de luz son una representación inherente al vídeo digital y a ciertos dispositivos de visualización 3D, especialmente los volumétricos y los autoestereoscópicos.

Una característica de los modelos del campo de luz es su elevado coste de almacenamiento y sus altos requisitos de memoria en el sistema de visualización. Para reducir este problema presentamos técnicas como la multirresolución y la visualización foveal. También demostramos que nuestros modelos soportan niveles de detalle continuos. Para asegurar un mínimo de interactividad durante la visualización del modelo, guardamos un conjunto de trabajo (*working set*) de imágenes en la GPU. Después cargamos el resto de imágenes bajo demanda hasta alcanzar el nivel de detalle deseado.

En este artículo presentamos un modelo de representación

de campos de luz que combina todas las características anteriores. Se ha probado con campos de luz de alta definición, de hasta 81920 imágenes de  $512 \times 512$  píxeles. El modelo también permite guardar información geométrica asociada a las imágenes del modelo. Nuestros experimentos demuestran que así se reducen los requisitos de almacenamiento entre 8 y 10 veces sin afectar a la calidad de los resultados. Además, el uso de información geométrica permite combinar varios campos de luz y geometría en una única escena.

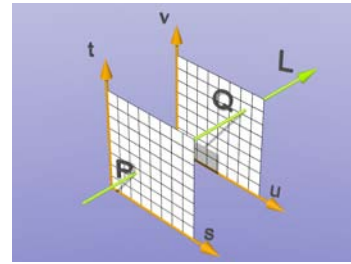
Los resultados confirman que este modelo de representación produce imágenes interactivas de alta calidad mientras que, para ser generadas, estas imágenes requieren entre medio y dos minutos de ejecución de un visualizador avanzado. Estos resultados se pueden aplicar a la generación, almacenamiento, compresión, transmisión, recuperación y visualización de imágenes 3D y 4D para dispositivos espaciales, como volumétricos y autostereoscópicos, y para televisión 3D. Estos medios de visualización permiten a varios usuarios ver simultáneamente imágenes 3D correctas sin utilizar gafas u otros elementos intrusivos. En este artículo presentamos también un ejemplo de utilización de nuestros modelos en una pantalla autoestereoscópica.

La estructura del artículo es la siguiente. Empezamos revisando trabajos anteriores sobre campos de luz y dispositivos espaciales. Seguimos con la descripción de nuestro modelo de representación y sus algoritmos de visualización. A continuación vemos cómo introducir información geométrica en nuestro modelo, sus ventajas y cómo afecta al algoritmo de visualización. Luego demostramos que esta información permite visualizar simultánea y correctamente campos de luz y geometría. Finalmente explicamos cómo visualizar nuestros modelos en un dispositivo autoestereoscópico. En las dos últimas secciones presentamos nuestros resultados, conclusiones e ideas para el futuro.

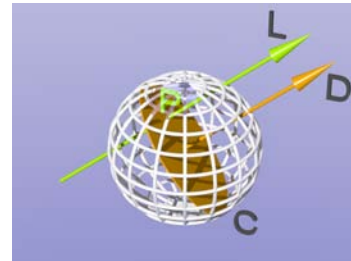
## 2. Antecedentes

Formalmente el campo de luz se define como una función que captura la radiancia  $R$  que fluye a través de todos los puntos  $(x, y, z)$  de una escena en todas las direcciones posibles  $(\theta, \phi)$  del espacio 3D [GGSC96, LH96]. Esta definición coincide con la de función plenóptica [AB91]. Si fijamos la longitud de onda y el tiempo, el campo de luz se convierte en una función escalar 5D  $R(x, y, z, \theta, \phi)$  que devuelve valores de radiancia para cada posición y dirección. El soporte de esta función son los rayos  $L(x, y, z, \theta, \phi)$  del espacio 3D.

Este soporte se reduce a 4D si se modela una región 3D libre de oclusores, pues sin ellos la radiancia no varía a lo largo de los rayos. Así el soporte de  $R$  pasa a ser el conjunto de líneas orientadas del espacio 3D. Para construir un modelo discreto del campo de luz hay que muestrear la radiancia. Esto requiere representar el espacio de líneas orientadas, cuya parametrización que puede ser o no estructurada.



(a) Parametrización planar.



(b) Parametrización esférica.

**Figura 1:** Dos parametrizaciones de las líneas que forman el soporte del campo de luz.

### 2.1. Modelos del campo de luz

Las parametrizaciones estructuradas muestrean los rayos siguiendo patrones regulares en los parámetros. Existen dos tipos de parametrizaciones estructuradas: planares y esféricas. Las planares parametrizan cada línea del campo de luz mediante dos puntos de intersección  $P$  y  $Q$  con dos planos paralelos (ver figura 1(a)). Así, la línea  $L$  tiene parámetros  $(s, t, u, v)$ . A cada par de planos se le conoce como *light slab* y típicamente se utilizan 6 para representar un objeto completo [LH96]. Esta parametrización fue inspirada originalmente por la holografía [HH92]. Gortler *et al.*, además, utilizan *proxies* geométricos para mejorar la reconstrucción de la función continua de radiancia [GGSC96]. Por último, Sloan *et al.* estudiaron la relación entre el número de muestras, el número de *slabs*, la velocidad de visualización y la calidad de las imágenes para campos de luz planares [PPS97].

Las parametrizaciones estructuradas esféricas son de dos tipos, la de las dos esferas y la de dirección y punto [CLF98]. Ambas tienen la ventaja de que producen un muestreo (casi)uniforme del espacio de líneas sobre el que se define el campo de luz. La figura 1(b) muestra la parametrización de dirección y punto, donde el rayo  $L$  viene dado por su dirección  $D$  y su intersección  $P$  con el plano soporte del gran círculo  $C$ . En este trabajo utilizamos esta parametrización, que se describe con detalle en la sección 3.

Las parametrizaciones no estructuradas no restringen los puntos de vista desde donde se capturan las imágenes del objeto. Varios autores han propuesto técnicas basadas en mode-

los geométricos aproximados [BBM\*01], cierto tipo de información de profundidad [SVSG01] y una medida del enfoque de los puntos del objeto [TN05].

## 2.2. Visualización espacial

Como el campo de luz representa la luz que pasa por cada punto 3D en todas direcciones, los campos de luz están indicados para producir imágenes 3D en dispositivos espaciales. La visualización espacial permite a varios usuarios observar simultáneamente una imagen 3D verdadera sin necesidad de utilizar gafas. Los campos de luz almacenan conjuntos de imágenes correspondientes a todas las vistas mostradas por un dispositivo espacial.

Por ejemplo, Isaksen *et al.* proponen un modelo reparametrizable del campo de luz [IMG00]. Es un modelo que permite extraer imágenes dependientes del punto de vista para componer imágenes integrales del campo de luz. Isaksen *et al.* demuestran cómo estas imágenes integrales se pueden visualizar mediante lenticulares, un tipo de dispositivo espacial. Los lenticulares utilizan una matriz de lentes que desvían la luz procedente de la pantalla de forma que distintos observadores reciban imágenes distintas en cada ojo.

Una tecnología similar utiliza varios proyectores para proyectar múltiples vistas sobre una pantalla lenticular [YCH\*05]. Para estas pantallas existen sistemas de captura de vídeo integral que graban vídeo de alta resolución mediante un conjunto esférico de cámaras [KOK\*06]. El problema de los lenticulares es que ofrecen muy pocas vistas, menos de diez típicamente. Para evitar este problema existen las pantallas holográficas retroproyectadas [BFB\*05]. Su coste es elevado, porque utilizan decenas de proyectores, pero permiten visualizar cerca de 45 vistas, cinco veces más que una pantalla lenticular.

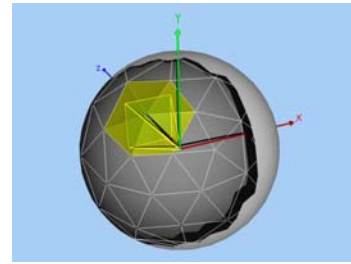
Otros dispositivos espaciales usan barreras de paralaje formadas por dos LCD superpuestos [PPK00]. El anterior visualiza tiras verticales alternativas negras y transparentes. El posterior visualiza un par estereó de manera que cada ojo ve una imagen distinta a través de las tiras del LCD anterior. Estos dispositivos, aunque no requieren el uso de gafas especiales, precisan de algún sistema de localización del usuario.

Finalmente, los dispositivos volumétricos muestran simultáneamente varias vistas de un modelo, proyectándolo sobre una superficie giratoria [FNH\*02] y [JMY\*07]. El sistema se encarga de sincronizar la rotación y la correcta proyección de las distintas vistas.

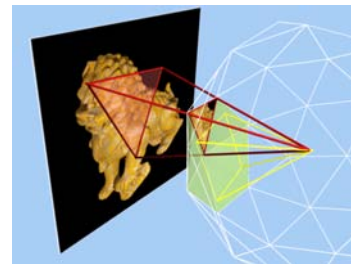
A continuación presentamos una implementación de campos de luz compatible con estos dispositivos.

## 3. Modelado y visualización de campos de luz

Para modelar y visualizar campos de luz hemos desarrollado un sistema que soporta modelos planares, esféricos y



(a) El volumen de la vista se sitúa dentro de la esfera seleccionando los triángulos visibles (en amarillo).



(b) Las coordenadas de textura de cada triángulo se obtienen proyectando sus vértices sobre el plano donde está la imagen del campo de luz (en rojo).

**Figura 2:** Algoritmo de visualización de un campo esférico.

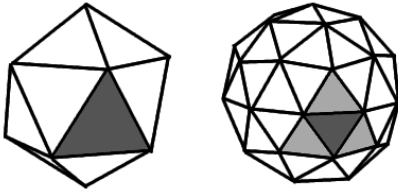
no estructurados [BEA\*08]. El sistema abstrae las características comunes a los tres tipos y define una interfaz simplificada para utilizarlo. Para la visualización estereoscópica de objetos aislados hemos comprobado que el mejor modelo es el esférico. Esto es debido a que ofrece una representación que no depende de los parámetros de la cámara. Los modelos planares son más adecuados para puntos de vista situados a un lado del objeto, mientras que los modelos no estructurados son adecuados para modelos capturados mediante cámaras de vídeo de tipo doméstico que se mueven libremente en el mundo 3D.

En esta sección describimos el modelo de representación de los campos de luz esféricos y sus características de multi-resolución, nivel de detalle continuo y visualización foveal.

### 3.1. Modelo de representación esférico

En esta representación cada rayo se parametriza por su dirección  $D = (\theta, \phi)$  y por su intersección  $P = (u, v)$  con un plano ortogonal a  $D$  (ver figura 1(b)). Para construir un campo de luz a partir de un modelo geométrico, éste se escala y se sitúa ajustado dentro de la esfera unidad. Luego se muestran las líneas que atraviesan dicha esfera y sus radiancias, obteniéndose así las imágenes del campo de luz [CLF98].

Para construir este modelo es necesario discretizar las di-



**Figura 3:** División del espacio de direcciones 3D (cada triángulo representa un haz): izquierda, división inicial basada en el icosaedro; derecha, primera subdivisión calculada al dividir en cuatro los triángulos de la izquierda.

recciones  $(\theta, \phi)$  del espacio 3D. Esto se hace tomando puntos sobre la superficie de la esfera unidad. Se parte de una triangulación de la esfera unidad y se toman como muestras direccionales los centros de los triángulos. Cada muestra aproxima el haz de direcciones que, partiendo del centro de la esfera, atraviesa la superficie del triángulo.

Para muestrear las líneas del soporte del campo de luz se discretizan todas las líneas que son paralelas a la muestra direccional e intersectan la esfera. En la figura 1(b) estas líneas son las que atraviesan el gran círculo  $C$ . En la práctica, las radiancias del campo de luz se obtienen calculando una proyección ortográfica del objeto a lo largo de cada muestra direccional. Cada imagen así obtenida se asocia al haz de direcciones que aproxima la muestra y se guarda como una textura asociada al triángulo que la representa.

El algoritmo de visualización utilizado es una adaptación del algoritmo del Lumigraph [GGSC96]. Este algoritmo está ilustrado en la figura 2. Inicialmente, se coloca el volumen de la vista dentro de la esfera teselada y se calculan los haces que intersectan con él (si es necesario se escalan la esfera o el volumen quedando ambos centrados en el origen). Los haces intersectados corresponden a los triángulos de la esfera visibles en esa vista (ver figura 2(a)). A continuación, cada triángulo se dibuja texturado con la imagen asociada a ese triángulo en el campo de luz. Las coordenadas de textura se calculan proyectando sus vértices sobre el plano soporte de la textura (en rojo en la figura 2(b)). La figura 5(a) muestra un campo de luz visualizado de este modo con los triángulos dibujados en alámbrico sobre la imagen.

### 3.2. Modelo multirresolución

Para construir un modelo multirresolución consideramos separadamente la dependencia posicional y la direccional del campo de luz. El muestreo posicional de la radiancia se hace tomando proyecciones ortográficas que se almacenan como texturas. Para soportar multirresolución en este espacio las texturas se guardan como *mipmaps*.

En el espacio direccional construimos un muestreo casi

uniforme partiendo del icosaedro como triangulación inicial (figura 3 izquierda). A continuación, dividimos cada uno de los 20 triángulos en cuatro subtriángulos y proyectamos sus nuevos vértices sobre la superficie de la esfera (figura 3 derecha). Esto genera 80 triángulos y 80 muestras que pasan por sus respectivos centros. Este proceso se aplica recursivamente para obtener 320, 1280, etc. muestras. Cuando se visualiza un modelo con pocas muestras direccionales aparecen costuras entre los triángulos dibujados por el visualizador (ver la cabeza del gárgola en la figura 4(a)). Con nuestra representación estos problemas se pueden eliminar usando interpolación o niveles de detalle más altos.

### 3.3. Nivel de detalle continuo

La discretización anterior permite una representación multirresolución lógica y sencilla. Sin embargo produce *poping* durante los cambios del nivel de detalle. Este problema se puede reducir o eliminar mediante transiciones suaves entre niveles de detalle, tanto en el dominio posicional como en el direccional. Para el suavizado en el dominio posicional una simple interpolación entre niveles de *mipmap* consecutivos es suficiente. En el dominio direccional se realiza una interpolación lineal entre el triángulo asociado a una determinada muestra y los triángulos asociados a sus hijos, usando  $\alpha$ -blending. La figura 4 muestra un ejemplo.

### 3.4. Visualización foveal

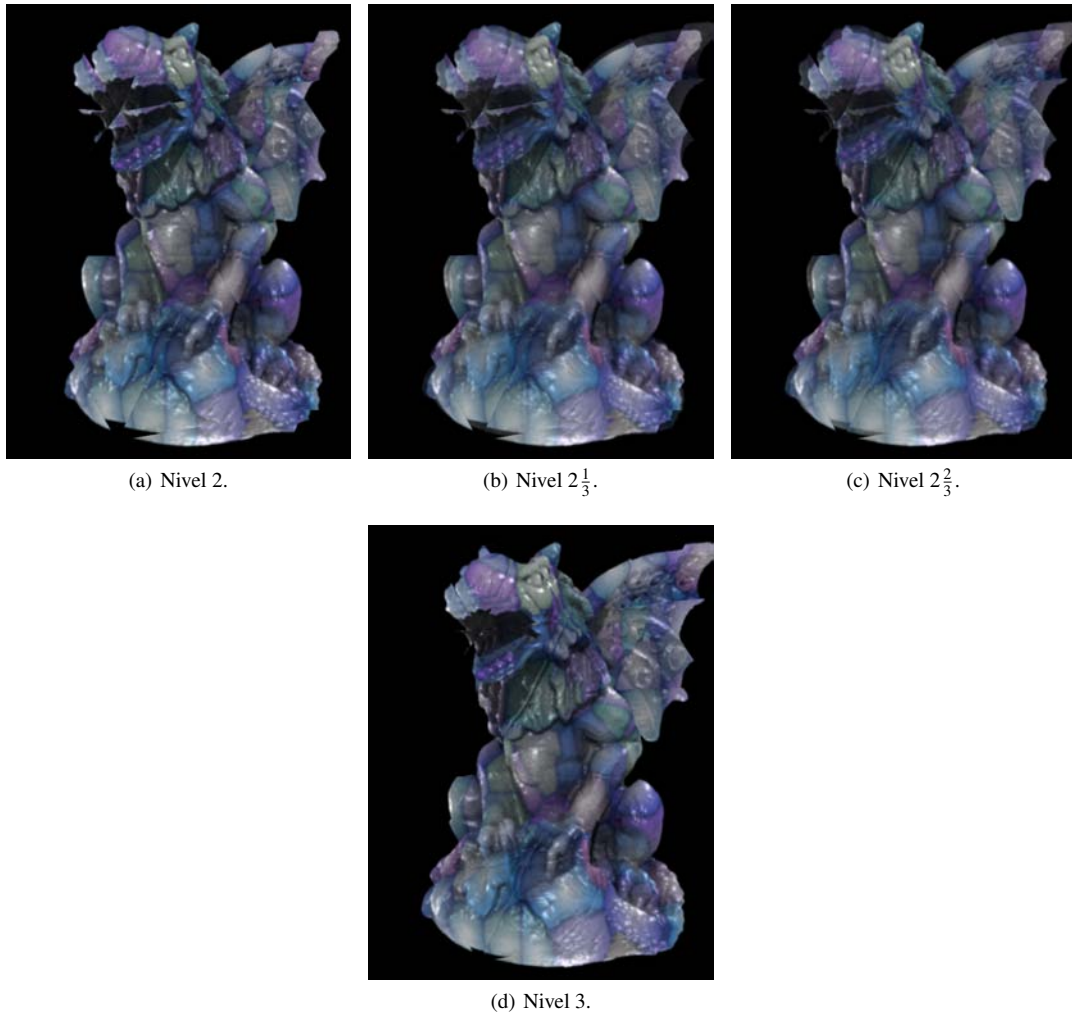
Nuestro sistema también soporta la visualización foveal reduciendo el nivel de detalle en zonas alejadas del centro de la pantalla [AR03]. Esto incrementa la velocidad de visualización afectando a la calidad sólo en la periferia de la imagen. Dado un ángulo que define el tamaño de la zona foveal, el algoritmo calcula el ángulo que forman la dirección de la cámara y la dirección de cada haz. Si este ángulo es menor que el foveal, el triángulo que representa al haz se subdivide y se utilizan sus cuatro hijos en su lugar. Si no, se visualiza el triángulo sin subdividirlo.

De este modo los triángulos fuera de la zona foveal se visualizan a menor nivel de detalle (ver figura 5). Esta técnica se puede extender a más de una área foveal y se puede aplicar a pantallas de gran tamaño, de 50 o más pulgadas, y a salas de cine de tipo estadio e IMAX. En estas pantallas el usuario fija su atención en el centro sin notar que cerca de los bordes la resolución de la visualización es menor.

## 4. Uso de información de profundidad

Esta información permite reducir substancialmente el coste de almacenamiento del campo de luz sin afectar a la calidad de su visualización. También dota a estos modelos de la funcionalidad necesaria para integrarlos como una primitiva más en un sistema de visualización convencional [DEA\*06].

La información de profundidad puede ser de dos tipos: Un



**Figura 4:** Campo de luz visualizado con niveles de detalle continuos 2,  $2\frac{1}{3}$ ,  $2\frac{2}{3}$  y 3.

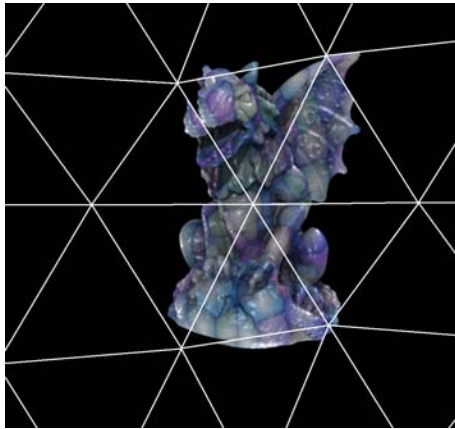
*proxy* geométrico que aproxima el objeto real mediante un modelo poligonal simplificado, o mapas de profundidad que asocian un valor de profundidad a cada muestra del campo de luz. Nuestra representación almacena un mapa de profundidad por cada imagen que compone el campo de luz. Para cada píxel de la imagen el mapa guarda la distancia (normalizada a  $[0, 1]$ ) desde la superficie del objeto al plano de la imagen. Sin compresión esto incrementa el espacio necesario entre un 50% y un 100% dependiendo de la resolución de los píxeles y las profundidades. No obstante, nuestros experimentos demuestran que las muestras direccionales se pueden reducir hasta 16 veces (2 niveles de detalle) sin afectar a la calidad de la visualización.

La figura 6 demuestra claramente la ventaja de utilizar el modelo de campo de luz con profundidad. El campo de luz usado para generar la imagen de la izquierda tiene 16 veces

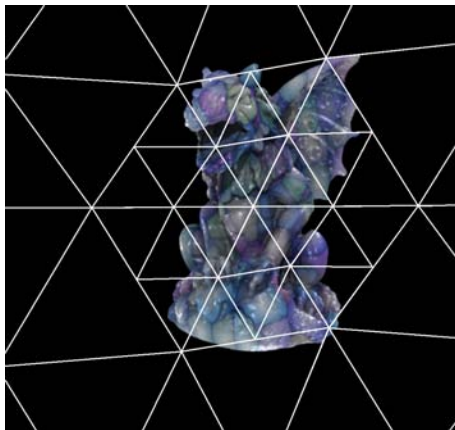
más imágenes que el campo de luz usado para la imagen de la derecha. Comparando ambas imágenes, se puede apreciar que la calidad es similar. De hecho, al ampliar ambas imágenes, se comprueba que la imagen de la izquierda es más borrosa. La imagen central muestra el resultado de utilizar el mismo campo de luz que la imagen derecha, pero sin profundidad. En ella se ve que no hay suficiente información para reconstruir adecuadamente el campo de luz.

Para comparar los requisitos de almacenamiento de las dos mejores opciones, hay que tener en cuenta el tamaño de las imágenes y el número de imágenes utilizadas por cada campo de luz. Para el campo de luz con profundidad, cada imagen puede ocupar hasta el doble de lo que ocupa una imagen del modelo sin profundidad. Aparte de la información de color (24 o 32 bits), se almacena la profundidad (8, 24, 32 o típicamente, 16 bits). Por otra parte, para el campo de luz sin





(a) Sin visualización foveal.



(b) Con visualización foveal.

**Figura 5:** Ejemplo de visualización foveal de un campo de luz: Los triángulos se subdividen más dentro del área foveal.

profundidad, hacen falta 16 veces más imágenes (esta vez, sin información de profundidad) que para el campo de luz con profundidad. Teniendo en cuenta dichos tamaños, se deduce que el campo de luz con profundidad necesita alrededor de la décima parte de espacio que el modelo sin profundidad.

#### 4.1. Visualización con mapas de profundidad

Este algoritmo modifica al anterior porque dibuja los triángulos sobre la superficie del objeto a representar. Para ello interseca los rayos que pasan por los vértices de cada triángulo con una reconstrucción 3D del mapa de profundidad correspondiente (ver figura 7). Dado un rayo, las coordenadas de su intersección se utilizan para calcular de forma más precisa las coordenadas de textura. Estas coordenadas también se utilizan para calcular la posición aproximada del triángulo sobre la superficie del objeto. En la práctica esto implica una reconstrucción de dicha superficie.

Esta reconstrucción tiene los siguientes problemas: ¿Qué ocurre cuando las profundidades asociadas a los vértices de un triángulo son muy dispares? ¿Y cuando uno de los rayos no interseca con el objeto a visualizar? En estos casos se divide recursivamente el triángulo. Esta subdivisión se controla mediante tres valores de tolerancia: El primero obliga a dividir todos los triángulos que ocupan más de un número de píxeles, el segundo obliga a dividir los triángulos cuyos vértices tienen profundidades asociadas demasiado dispares y el tercero limita el proceso de subdivisión impidiendo que se subdividan triángulos que ocupen menos de un píxel.

Nuestros experimentos indican que una primera tolerancia de 40 píxeles produce buenos resultados. Valores más pequeños requieren muchas subdivisiones e incrementan los tiempos sin mejorar los resultados. Para la segunda tolerancia utilizamos 0.05 dentro del rango normalizado de profundidades  $[0, 1]$ . Este valor permite perfilar mejor los bordes del objeto que no pertenecen a la silueta. Finalmente, la última tolerancia se fija en un píxel porque no tiene sentido dibujar triángulos más pequeños y porque valores más grandes producen *mordiscos* en las siluetas de los objetos que aumentan de tamaño con la tolerancia.

Las imágenes de este artículo se han generado con un visualizador configurado con los valores de tolerancia anteriores. Además, utilizamos profundidades para calcular más precisamente las coordenadas de textura de los triángulos. Esto reduce las costuras que, como en la figura 4(a), aparecen en las aristas compartidas por triángulos adyacentes. Finalmente, situar los triángulos sobre la superficie del objeto permite visualizar simultáneamente múltiples campos de luz y geometría.

#### 4.2. Integración de múltiples campos de luz y geometría

Para integrar varios campos de luz en una escena es necesario guardar más de una imagen por muestra direccional. Por ello, almacenamos para cada modelo una imagen situada a una distancia diferente a lo largo de cada dirección. Esta aproximación puede producir errores de visualización graves cuando los modelos se cruzan (ver figura 8).

Este problema se corrige usando la información de profundidad para calcular las posiciones exactas de los vértices de los triángulos a visualizar. Con esta información los triángulos se dibujan sobre la superficie del objeto original. Así, la visibilidad de la escena se resuelve correctamente.

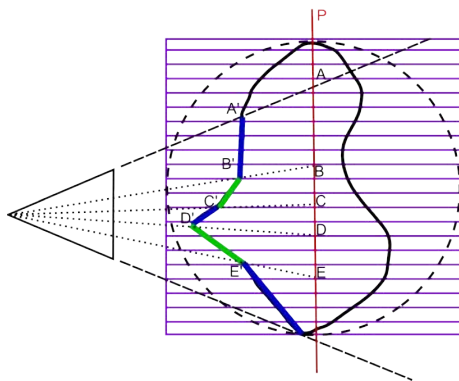
Este algoritmo permite también integrar campos de luz y geometría en una misma escena. De este modo el campo de luz puede ser tratado como una primitiva gráfica más de una escena geométrica. También permite integrar correctamente elementos geométricos, tales como etiquetas o anotaciones, en campos de luz.



**Figura 6:** Un modelo visualizado con las siguientes resoluciones direccionales: izquierda, 20480 sin profundidad; centro, 1280 sin profundidad; y derecha, 1280 con profundidad. Las dos primeras imágenes se han generado con interpolación, la tercera sin interpolación. En la imagen del centro se observan imágenes dobles (ghosting).

nombre	modelo geométrico		modelo del campo de luz				
	número de polígonos	tiempo por imagen	muestras direccionales	tiempo total	tamaño imágenes en disco	tamaño imágenes en memoria	tamaño mapas
gárgola	1726372	122 seg	1280	43h 22m	201 MB	1.25 GB	640 MB
guerrero	1752348	64 seg	1280	22h 45m	161 MB	1.25 GB	640 MB
león	1311956	48 seg	20480	11d 13h 52m	3.42 GB	20.00 GB	10.24 GB

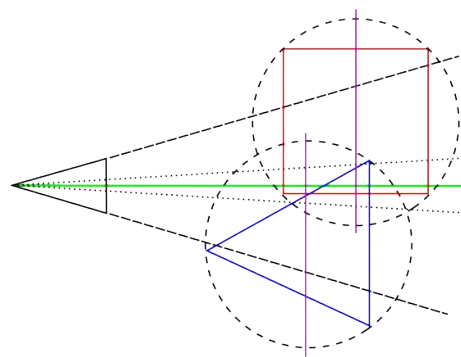
**Tabla 1:** Características de los modelos de este artículo: Los tiempos de generación son sobre un solo PC, los tamaños de los modelos son sin compresión, salvo en disco, donde las imágenes se guardan en formato PNG.



**Figura 7:** Analogía 2D de la aproximación con mapas de profundidad a la superficie del objeto: Durante el dibujado, se lanzan rayos desde el punto de vista a través de los vértices de los triángulos; en vez de intersectarlos con el plano P (puntos A, B, C, ...), se intersectan con el mapa de profundidad (puntos A', B', C', ...).

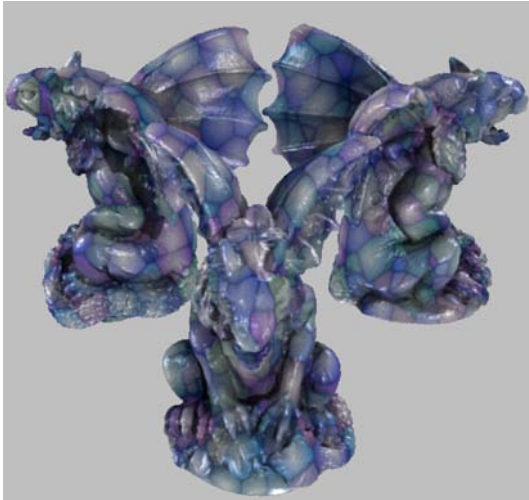
## 5. Resultados

Para generar los campos de luz mostrados en este artículo hemos usado tres modelos geométricos del proyecto



**Figura 8:** A lo largo de la visual verde el cubo está más cerca que el tetraedro; sin embargo, el plano de la imagen que lo representa está detrás del que representa al tetraedro. En la imagen final el tetraedro tapará erróneamente al cubo.

AIM@Shape, así como los programas Blender y Yafaray. Los modelos, llamados gárgola, guerrero y león, se muestran en la figura 9(b) y sus características en la tabla 1. Para generar las imágenes de los campos de luz se han utilizado *Ambient Occlusion*, iluminación especular de Cook-Torrance y sobremuestreo de 8 muestras por píxel. Los modelos gene-

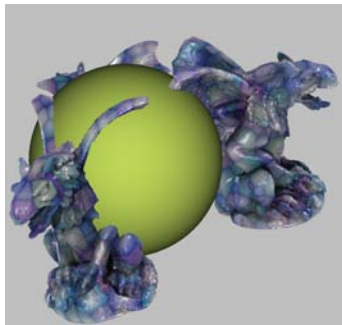


(a) Escena 1: Tres instancias de un modelo con profundidad y visibilidad resueltas correctamente.

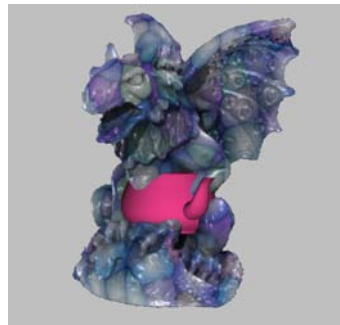


(b) Escena 2: Escena con cuatro modelos visualizados correctamente.

**Figura 9:** Visualización simultánea de múltiples modelos del campo de luz.



(a) Escena 3: Dos modelos del campo de luz y una esfera.



(b) Escena 4: Una tetera intersecta un modelo del campo de luz.



(c) Escena 5: Dos modelos del campo de luz interpenetrantes.

**Figura 10:** Estas escenas con modelos geométricos y del campo de luz demuestran cómo la visibilidad se resuelve correctamente.

rados tienen  $512 \times 512$  píxeles de resolución posicional y 1280 direcciones de resolución direccional, excepto el león que tiene 20480 direcciones. La generación se ha hecho sobre una granja de hasta 100 ordenadores pertenecientes a los laboratorios docentes de nuestro departamento. El número de polígonos se ha limitado para que sólo cuesten unos días de generar y así hacerlo en periodos no lectivos. La tabla 1 contiene los tiempos de generación (suponiendo un solo PC) y el tamaño de cada modelo.

En la página <http://www.sig.upv.es/ALF/papers/ceig2008> se pueden encontrar vídeos que demuestran cómo se visualizan nuestros campos de luz en dispositivos convencionales y autoestereoscópicos.

### 5.1. Visualización monoscópica

Para visualizar los campos de luz hemos utilizado un Intel Core 2 Quad a 2.4Ghz con 2GB de memoria RAM y una tarjeta gráfica NVIDIA GeForce 8800 GT con 512 MB de memoria. La figura 9 muestra una escena con múltiples campos de luz. Para modelar la escena de la figura 9(a) se instancia tres veces el modelo de la gárgola. La caché del sistema permite compartir las mismas imágenes en todas las instancias del mismo campo de luz. La figura 9(b) muestra dos copias del modelo de la gárgola junto al león y el guerrero. En la figura 10 se aprecia cómo se resuelve correctamente la visibilidad entre objetos geométricos y campos de luz, así como entre campos de luz diferentes. Por último, la Tabla 2



contiene el tamaño de estas escenas y su velocidad de visualización.

Esto demuestra que podemos integrar varios modelos con geometría en una única escena, resolviendo correctamente la visibilidad. También corrobora que nuestros modelos son competitivos frente a la visualización interactiva de objetos complejos mediante GPU. Cuando la geometría y la iluminación son tan complejas que no es posible visualizar el objeto en tiempo real con el hardware disponible actualmente, es mejor generar su campo de luz fuera de línea y visualizarlo en tiempo real usando la GPU. Además futuras mejoras en las GPU acelerarán la visualización tanto de modelos geométricos complejos, como de nuestros modelos del campo de luz.

## 5.2. Visualización autoestereoscópica

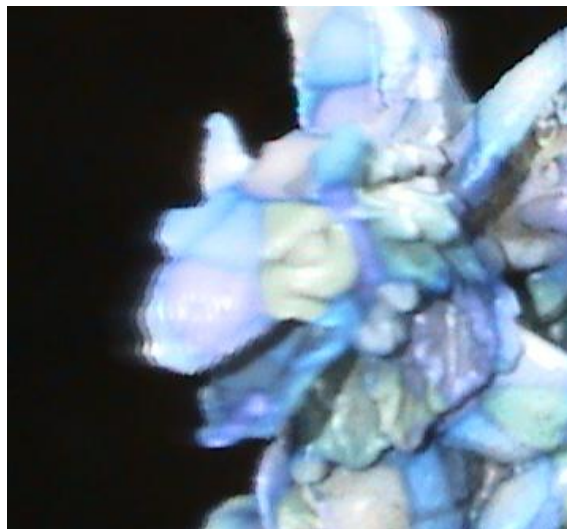
En la actualidad los dispositivos de visualización espacial admiten dos posibles tipos de entrada. El primero es una descripción geométrica de la escena en un formato estándar como, por ejemplo, OpenGL. Estos dispositivos disponen de *wrappers* o *drivers* adaptados que generan internamente las vistas necesarias para la visualización autoestereoscópica. El segundo tipo de entrada es un conjunto de imágenes, una por cada vista que el dispositivo es capaz de mostrar.

Los campos de luz son adecuados para este tipo de dispositivos, ya que permiten obtener fácilmente las vistas necesarias a partir de las imágenes precaculadas que almacenan. Sin embargo, muy pocas implementaciones pueden generar geometría a partir de campos de luz. Nuestro sistema es capaz de ambas cosas y por tanto puede utilizar las dos tecnologías de dispositivos de visualización autoestereoscópica descritas.

Para demostrar la visualización autoestereoscópica de nuestros modelos hemos utilizado un monitor autoestereoscópico Philips WOWvx de 20". Este monitor utiliza tecnología lenticular de 9 vistas con paralaje horizontal y una pantalla táctil [BP00]. Además permite dos modos de funcionamiento: visualización de modelos OpenGL y visualización de imágenes con su correspondiente información de profundidad. El resultado son imágenes integrales formadas por 9 vistas que cubren una anchura de campo reducida. La figura 11 muestra el detalle de una fotografía del dispositivo en funcionamiento. Debido a la apertura de la lente de la cámara con la que se tomó la instantánea, se aprecian imágenes dobles que corresponden a varias de las vistas generadas por el dispositivo.

## 6. Conclusiones y trabajo futuro

Hemos presentado un modelo de campo de luz esférico con características de multiresolución, niveles de detalle continuos y visualización foveal. El modelo soporta la visualización interactiva de objetos complejos a mayor velocidad



**Figura 11:** *Detalle de una fotografía de un campo de luz en la pantalla autoestereoscópica: Se aprecia un efecto de imagen doble debido a que son visibles varias de las 9 vistas.*

que lo que costaría su visualización en tiempo real con el hardware actual.

Además hemos ampliado el modelo con información de profundidad. Esta información reduce entre 8 y 10 veces el coste de almacenamiento de los modelos sin afectar a la calidad de su visualización. También demostramos por primera vez que podemos visualizar escenas con múltiples campos de luz y geometría integrados, resolviendo correctamente la visibilidad.

Finalmente, hemos introducido la visualización autoestereoscópica para nuestros modelos del campo de luz. Los resultados demuestran que estos modelos son adecuados para pantallas lenticulares y que 9 vistas son insuficientes para proporcionar una buena sensación 3D a múltiples usuarios. Estamos investigando la utilización de otros dispositivos no lenticulares con un número mayor de vistas.

También estamos trabajando en la implementación de campos planares y no estructurados. Concretamente, queremos capturar campos con una videocámara no profesional para luego convertirlos a una representación estructurada y potencialmente no uniforme. Para ello trabajamos en modelos no uniformes y en algoritmos de conversión.

## Agradecimientos

Queremos dar las gracias a los revisores por sus comentarios y sugerencias. También queremos agradecer a los técnicos y a la dirección de nuestro Departamento la cesión de sus laboratorios docentes, con más de 100 ordenadores, para generar los campos de luz mostrados en este artículo. José Luis Hidalgo nos ayudó a generar dichos campos.

escena	descripción	tamaño en disco	tamaño en memoria	velocidad
escena 1	tres gárgolas formando un círculo	841 MB	1,87 GB	4,2 FPS
escena 2	dos gárgolas, un león y un guerrero	2,44 GB	5,62 GB	5,3 FPS
escena 3	tres gárgolas y una esfera	841 MB	1,87 GB	4,2 FPS
escena 4	gárgola con tetera incrustada	841 MB	1,87 GB	32 FPS
escena 5	león y gárgola interpenetrantes	974 MB	3,75 GB	13 FPS

**Tabla 2:** Características de las escenas de las figuras 9 y 10; el modelo del león sólo utiliza 1280 muestras direccionales de las 20480 disponibles.

El modelo geométrico del león (Chinese Dragon) es cortesía del INRIA. Los modelos de la gárgola (Gargoyle) y el guerrero (Grog) son cortesía del VCG-ISTI. Los tres se obtuvieron del repositorio del proyecto AIM@SHAPE. Por último, nuestro trabajo ha sido financiado en parte por becas y ayudas del Vicerrectorado de Investigación de la Universidad Politécnica de Valencia y por el proyecto TIN2005-08863-C03-01 del Ministerio de Ciencia e Innovación.

## References

- [AB91] ADELSON E. H., BERGEN J. R.: The plenoptic function and the elements of early vision. *Computational Models of Visual Processing* (1991), 3–20.
- [AR03] ASHDOWN M., ROBINSON P.: The escritoire: A personal projected display. In *Proceedings of WSCG'2003* (2003), pp. 33–40.
- [BBM\*01] BUEHLER C., BOSSE M., MCMILLAN L., GORTLER S., COHEN M.: Unstructured lumigraph rendering. In *SIGGRAPH '01* (2001), pp. 425–432.
- [BEA\*08] BLASCO J., ESCRIVÁ M., ABAD F., QUIRÓS R., CAMAHORT E., VIVÓ R.: A generalized light-field API and management system. In *Proceedings of WSCG'2008* (Plzen – Bory, Czech Republic, 2008).
- [BFB\*05] BALOGH T., FORGÁCS T., BALET O., BOUVIER E., BETTIO F., GOBBETTI E., ZANETTI G.: A scalable holographic display for interactive graphics applications. *IEEE VR 2005 Workshop on Emerging Display Technologies* (March 2005).
- [BP00] BERKEL C. V., PARKER D. W.: Austostereoscopic display apparatus. U.S. Patent No. 6,118,584, 2000.
- [CLF98] CAMAHORT E., LERIOS A., FUSSELL D.: Uniformly sampled light fields. In *Proc. Eurographics Rendering Workshop '98* (1998), pp. 117–130.
- [DEA\*06] DOMINGO A., ESCRIVÁ M., ABAD F., R. VIVÓ, E. CAMAHORT: Introducing extended and augmented light fields for autostereoscopic displays. In *Proceedings of 3rd Ibero-American Symposium in Computer Graphics* (2006), pp. 64–67.
- [FNH\*02] FAVALORA G. E., NAPOLI J., HALL D. M., DORVAL R. K., GIOVINCO M. G., RICHMOND M. J., CHUN W. S.: 100 million-voxel volumetric display. In *Proceedings of the SPIE* (2002), vol. 4712, pp. 300–312.
- [GGSC96] GORTLER S. J., GRZESZCZUK R., SZELISKI R., COHEN M. F.: The lumigraph. In *Proc. SIGGRAPH '96* (1996), pp. 43–54.
- [HH92] HAINES K., HAINES D.: Computer graphics for holography. *IEEE Computer Graphics and Applications* (January 1992), 37–46.
- [IMG00] ISAKSEN A., MCMILLAN L., GORTLER S. J.: Dynamically reparameterized light fields. In *Proc. SIGGRAPH '00* (2000), pp. 297–306.
- [JMY\*07] JONES A., MCDOWALL I., YAMADA H., BOLLAS M., DEBEVEC P.: An interactive 360° light field display. In *SIGGRAPH '07* (2007), ACM.
- [KOK\*06] KOIKE T., OIKAWA M., KIMURA N., BENIYAMA F., MORIYA T., YAMASAKI M.: Integral videography of high-density light field with spherical layout camera array. In *Proceedings of SPIE* (2006).
- [LH96] LEVOY M., HANRAHAN P.: Light field rendering. In *Proc. SIGGRAPH '96* (1996), pp. 31–42.
- [PPK00] PERLIN K., PAXIA S., KOLLIN J. S.: An autostereoscopic display. In *Proc. SIGGRAPH '00* (2000), pp. 319–326.
- [PPS97] PETER-PIKE SLOAN MICHAEL F. COHEN S. J. G.: Time critical lumigraph rendering. In *Proc. 1997 Symposium on Interactive 3D Graphics* (1997).
- [SVSG01] SCHIRMACHER H., VOGELGSANG C., SEIDEL H., GREINER G.: Efficient free form light field rendering. In *Proceedings of Vision, Modeling, and Visualization 2001* (2001).
- [TN05] TAKAHASHI K., NAEMURA T.: Unstructured light field rendering using on-the-fly focus measurement. *ICME 2005. IEEE International Conference on Multimedia and Expo* (July 2005), 205–208.
- [YCH\*05] YANG R., CHEN S., HUANG X., LI S., WANG L., JAYNES C.: Towards the light field display. *IEEE VR 2005 Workshop on Emerging Display Technologies* (March 2005).