

# Visualización mediante punteado de monumentos y restos arqueológicos escaneados

D. Martín, G. Arroyo y M.V. Luzón  
GIIG, Universidad de Granada

---

## Abstract

*Una de las nuevas áreas de aplicación de los escáneres es la captura de monumentos y restos arqueológicos para el mantenimiento del patrimonio cultural, siendo la documentación una de las tareas más importantes al trabajar con monumentos, la cual suele estar acompañada de imágenes de los mismos. Tradicionalmente estas imágenes han sido ilustraciones hechas a mano utilizando técnicas de plumilla y punteado. En este trabajo se presenta un método de obtención de ilustraciones mediante punteado a partir de monumentos y restos arqueológicos escaneados. Frente a trabajos previos, la solución que se presenta es simple y rápida, no necesitando tratamientos costosos de las nubes de puntos, permitiendo al usuario un ajuste interactivo así como la generación de animación manteniendo la coherencia entre fotografías.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling, Hierarchy and geometric transformations

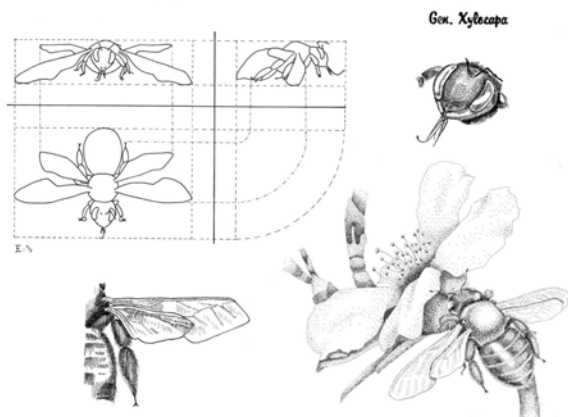
---

## 1. Introducción

En las aplicaciones de patrimonio cultural el escaneado se está convirtiendo en una herramienta básica para arqueólogos, restauradores, museólogos, etc. Mediante el escáner se capturan los modelos, desde edificios completos hasta pequeños pedazos de una pieza arqueológica, de manera rápida y eficiente. De esta forma se obtiene un modelo virtual del objeto, el cual permite la aplicación de una gran cantidad de procesos y tareas productivas. Por ejemplo, se pueden planificar tareas de restauración o recuperación, realizar reconstrucciones virtuales, estudios de fatiga sobre materiales, etc.

Una de las principales tareas en los trabajos arqueológicos es el de la documentación. Es normal, por la naturaleza de los trabajos, que los mismos se acompañen de imágenes de los monumentos o restos arqueológicos. Si bien se ha podido hacer uso de la fotografía, tradicionalmente ha habido una amplia utilización de la ilustración manual [Hod89], al igual que en otras ciencias como la entomología, (Véase Figura 1), ornitología, botánica, etc. El motivo de su amplio uso frente a la fotografía se debe a que la imagen producida por el artista sufre un procesamiento o interpretación que hace que se muestre la información más relevante de la forma más sencilla y fácil de entender. Esto contrasta frente al detalle físico-realista de una fotografía.

Aunque existen numerosas técnicas, las más usadas en arqueología son aquellas que se basan en la ilustración mediante plumilla y punteado, esto es, la distribución de puntos de tamaño constante de tal forma que transmita la información de forma y tono. Las imágenes realizadas con tinta y plumilla sobre papel blanco poseen varias ventajas, entre las que se incluyen la posibilidad de representar la forma y el tono de una manera muy económica y expresiva. La ventaja económica se consigue cuando el trabajo tiene que ser impreso pues sólo necesita del uso de tinta [Woo94]. La componente expresiva implica que la información mostrada es más fácil de tratar por el observador. Incluso haciendo visibles características que no se mostrarían en una imagen realista [WS94], el ilustrador es capaz de dirigir la atención hacia aquellas partes que considera más importantes dando un nivel mayor de detalle mediante un mayor número de puntos. Las zonas menos importantes quedan con una descripción más simple. Las imágenes con punteado presentan otra ventaja frente a las que se realizan con líneas y es su invariabilidad frente a la orientación. Por otra parte, en la ilustración con plumilla el tono se obtiene mediante un rayado. Este rayado tiene una dirección concreta en su definición y el cambio de la misma puede suponer un cambio en el tono o un efecto no deseado en la visualización.



**Figure 1:** Ejemplo de ilustración con plumilla y punteado realizado por Elena Piñar.

La idea que motivó este trabajo se basa en que la visualización del conjunto total de puntos de una toma de escáner se puede considerar como una versión “degenerada” de la imagen con punteado de un monumento, entendiendo por degenerada el que se dibujarían todos los puntos. Esto es, si dada una malla y un observador, a una distancia determinada, la malla se visualiza, se mostrará la forma del objeto escaneado como si el mismo estuviera hecho mediante punteado, con la salvedad de que no se distinguen zonas ni por importancia ni por tono. Partiendo de esta idea nos planteamos la posibilidad de buscar un método que filtrara los puntos de tal manera que permitiera salvar dichos inconvenientes.

Este trabajo aporta una solución al problema de la visualización mediante punteado de modelos representados mediante nubes de puntos. Para ello se han combinado de una manera nueva una serie de técnicas sencillas ya conocidas: interpolación, triangularización, registrado, simplificación y visualización. Mediante las mismas hemos obtenido una solución que resulta simple de implementar y cuyos resultados se pueden considerar como buenos o muy buenos\* tanto en imágenes estáticas como dinámicas, permitiendo la generación en tiempo real con modelos compuestos por cientos de miles de puntos.

## 2. Trabajos previos

La visualización expresiva\*\* es un área que ha conseguido un gran interés en los últimos años, dentro de los gráficos por ordenador. Mientras que en la generación de imá-

genes fotorrealistas el objetivo es obtener imágenes indistinguibles de las reales tomadas por una cámara, en el caso de la visualización expresiva se busca transmitir el “significado” de las cosas más que su apariencia, más el fondo que la forma [GG01, SS02]. En la visualización expresiva hay conocimiento e interpretación mientras que en la fotorrealista no.

Aunque existen numerosas áreas, nuestros trabajos se centran en la generación de imágenes con pluma y tinta. Un trabajo pionero fue el desarrollado por Winkenbach y Salesin [WS94]. En él se presentan una serie de técnicas para, a partir de un modelo tridimensional basado en superficies, obtener imágenes que simulan las realizadas a mano con plumilla. En general, después se han ido produciendo variantes en temas y objetivos de este trabajo original [WS96, M\*97, DS00, M\*00].

Dentro de las imágenes con pluma y tinta nos centraremos, concretamente en el punteado, normalmente realizado con una plumilla técnica tipo “rotring”. Las imágenes generadas con esta herramienta están conformadas por puntos del mismo tamaño, dispuesto de tal manera que permiten transmitir la idea de forma y tono, simplemente cambiando su posición y densidad. En nuestro trabajo usamos los puntos como primitiva geométrica y como primitiva de visualización [LW85, ZP\*01].

Entre los trabajos que crean las imágenes en el espacio de la imagen hay que citar el artículo de Deussen [DHvOS00], basado en un editor que genera medios tonos mediante puntos. Secord [Sec02], utiliza los diagramas centroidales de Voronoi para distribuir los puntos a partir de una imagen. Secord [SHS02], además, utiliza una variante del método anterior para dibujar puntos y otros tipos de primitivas usando funciones de probabilidad de densidad.

Yuan [YNZC05] utiliza un espacio intermedio entre la imagen y el modelo, “geometría-imagen”, para obtener distintas propiedades del modelo. Zakaria [ZS04] utiliza los puntos como primitiva de visualización y utiliza el resultado en el espacio de la imagen para obtener las siluetas como complementos al punteado. Pastor [PFS03] usa un procedimiento parecido al de el colapso de arista [Hop96] para el posicionamiento de los puntos, permitiendo la coherencia entre fotogramas. El trabajo de Xu [XC04] aplica una serie de pasos similar al método que aquí se presenta, pero con un grado de complejidad mucho mayor. En los trabajos de Lu [LME\*02, LM\*03] se hace una visualización de volúmenes mediante punteado haciendo un cálculo de gradientes en los vóxeles.

Maciejewski [MIA\*08] realiza un estudio sobre las diferencias entre las imágenes de punteado hechas a mano y con ordenador. Una de las indicaciones especialmente importantes que se han obtenido de dicho estudio es que, aunque ambos tipos de imágenes son relativamente fáciles de distinguir por los observadores, los “estilos” producidos por los huma-

\* Las imágenes obtenidas han sido evaluadas por la ilustradora Elena Piñar Toro

\*\* Preferimos el nombre del área dado por [LS95] en vez de no fotorrealista, ya que es una definición en positivo, no mediante la negación de un concepto

nos y por los ordenadores tenían el mismo grado de aceptación.

### 3. Tratamiento de los datos

En esta etapa se tratan los datos para permitir la visualización. El precálculo de muchas variables permite una gran aceleración de la visualización.

#### 3.1. Ajuste matricial de los datos

Una vez capturados los datos mediante un escáner láser, éstos deben sufrir una serie de procesos.

El primero es el de la lectura de los datos. Una de las posibilidades del software del escáner es la de generar ficheros en la que los datos son indexados siguiendo la secuencia de barrido propia del escáner, lo cual permite obtener una distribución matricial de los datos.

#### 3.2. Interpolación

El segundo proceso es el de la interpolación para obtener un número mayor de puntos. Esta etapa se aplicará o no, dependiendo de las necesidades del usuario y de las características del modelo. En general, las principales características que se buscan en una imagen con punteado son la forma y el tono. En nuestro modelo de datos escaneados, la forma es inherente, pero el tono no es contemplado ya que sólo se captura información geométrica (salvo la textura, que muestra unas condiciones de iluminación limitadas al momento de la toma).

Una de las características que tenemos que tener en cuenta es el tamaño de la imagen de salida, ya que no es lo mismo dibujar en un área de 500x500 píxeles que en una de 2000x2000 píxeles. Dado que contamos con un número de muestras prefijado y determinado por el escáner o por el usuario, tenemos que buscar alguna solución para el caso en el que el conjunto de puntos original no sea suficiente para cubrir las distintas zonas con una cierta densidad, dejando partes en blanco que no resultan naturales. Esto se resuelve aumentando el número de puntos mediante la interpolación de los datos. El procedimiento es trivial utilizando la estructura matricial de la que se parte. Así, si se tiene una matriz de  $m \times n$  puntos, el resultado será una de  $2m \times 2n$  puntos.

#### 3.3. Triangularización

El objetivo de la triangularización es el de facilitar la correcta visualización del modelo ya que al redibujar, las caras visibles eliminarán aquellos puntos que no lo sean. La estructura matricial de los puntos facilita mucho el proceso de triangularización. El procedimiento es muy sencillo, consistiendo en tomar los datos en columnas de dos en dos e ir recorriendo las posiciones por filas. Para cada paso consideramos 4 puntos. Entre los mismos se pueden generar dos

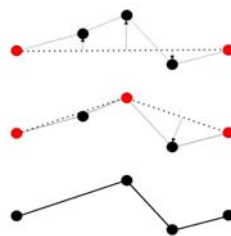


Figure 2: Procedimiento del algoritmo de Douglas-Peucker

triángulos. Para ello, lo único que tenemos que comprobar es que los tres vértices que se van a usar para crear la cara, son válidos (no pertenecen a datos nulos). Para cada triángulo que se crea se obtiene su normal.

#### 3.4. Cálculo de normales para los puntos

El cálculo de las normales para los puntos es necesario para poder implementar el tono, haciendo que unos puntos aparezcan o no según estén o no iluminados. Para dicho cálculo hacemos uso de las normales de los triángulos calculados en el paso previo, hallando el valor intermedio de las normales de las caras que comparten el punto. Esta información se usa en la etapa de visualización.

#### 3.5. Simplificación jerárquica de los puntos

Para generar una imagen con punteado a partir del conjunto total de puntos de la toma, es necesario seleccionar aquellos que permitan mostrar una imagen más realista (más parecida a la correspondiente realizada por un artista). Para ello es necesario realizar una clasificación de los puntos teniendo en cuenta que se van a extraer las características visuales más importantes. En el caso de los monumentos arqueológicos estas características suelen ser las zona angulares que representan esquinas y bordes. Esto, que en principio puede parecer una solución arbitraria, se relaciona profundamente con la geometría constructiva que usualmente utiliza el hombre en sus construcciones, basada en las líneas rectas horizontales y verticales, frente a las formas curvas que se muestran en la mayoría de los organismos vivos.

Una de las posibilidades consiste en el estudio de la disposición de las normales en los puntos, buscando aquellos en los que se produzcan grandes variaciones, pero se ha comprobado que no produce resultados del todo correctos.

La solución aportada se basa en el algoritmo de Douglas-Peucker [DP73]. El objetivo de este método es mantener aquellos puntos que tienen una mayor importancia en la forma de la curva, eliminando el resto. El algoritmo funciona de la siguiente manera. Dados los dos puntos extremos,  $P_0$  y  $P_n$  de la secuencia de  $n$  puntos que se van a simplificar  $P_1, P_2, P_3, \dots, P_{n-1}$  y dado un factor de distancia  $d$ , se

calculan, para cada punto, la distancia del punto a la recta  $P_0 - P_n$ , obteniendo  $d_1, d_2, d_3, \dots, d_{n-1}$ . Se calcula el máximo de estas distancias y se comprueba si es mayor que  $d$ . En caso afirmativo, el punto correspondiente a dicha distancia se mantiene y se convierte en lo que se denomina un “punto ancla”  $P_a$ , y se invoca recursivamente al procedimiento con  $P_0 - P_a$  y  $P_a - P_n$ . En caso contrario, se eliminan todos los puntos entre  $P_0$  y  $P_n$  (véase Figura 2).

Según el planteamiento descrito, y suponiendo un modelo que no cambia, cada vez que se ajusta el umbral es necesario aplicar el algoritmo. La mejora que se introduce consiste en aplicar el algoritmo de Douglas-Peucker al conjuntos de puntos, almacenando para cada punto su distancia a la recta que en ese momento se está usando. Al final, cada punto tiene la distancia a la recta en el último procesado que se le aplicó. Los puntos más importantes tendrán distancias más grandes mientras que los menos importantes tendrán distancias más pequeñas. De esta forma, realizamos el pre-cálculo del algoritmo de Douglas-Peucker, dejando el proceso de visualización reducido a realizar una búsqueda de los puntos que están por encima o por debajo de cierta distancia umbral.

Es importante reseñar que este algoritmo se desarrolló originalmente para la simplificación de polilíneas que delimitaban contornos geográficos en dos dimensiones. En nuestro trabajo, el algoritmo ha presentado un comportamiento excelente, para una gran variedad de escenarios y monumentos. Para poder usar este algoritmo, se hace uso de la estructura matricial en la que se han almacenado los puntos, tomándolos por filas o por columnas. Sólo hay que tener en cuenta si la polilínea es cerrada o abierta para considerar o no los puntos extremos. Si partimos de que el escáner ha hecho un barrido horizontal de 360 grados, entonces, y para evitar que sistemáticamente aparezcan los puntos inicial y final, es necesario utilizar los puntos anterior al primero y siguiente al último. Mediante este cruce de valores se tratan los puntos extremos de igual manera que el resto.

#### 4. Visualización

Con los datos obtenidos en las etapas anteriores se pasa a la visualización e interacción sobre el conjunto de datos, con el fin de producir las imágenes mediante punteado.

##### 4.1. Registrado de las mallas

El método expuesto hasta ahora funciona con una toma, pero si el monumento escaneado es grande o complejo, usualmente es necesario recurrir a una serie de tomas desde distintas posiciones para capturar la información que está en zona de sombra respecto a otras tomas. Los datos obtenidos están localizados con respecto al sistema de coordenadas del escáner, lo cual hace que no coincidan. Para ello es necesario recurrir al registrado. El procedimiento consiste en ir tomado de dos en dos las distintas tomas e ir ajustando uno de los sistemas de coordenadas al otro.

El procedimiento que hemos implementado utilizada una versión simplificada del algoritmo de Horn [Hor87], basado en tres puntos y utilizando matrices. Dada una toma, que llamaremos “fija” y otra, que denominaremos “flotante”, seleccionamos tres puntos de cada una de ellas. Estos seis puntos se deben relacionar por parejas, por lo que un punto en la toma fija debe relacionarse con su correspondiente en la toma flotante. Dadas las características del escaneado, en raras ocasiones las nubes de puntos de dos tomas son coincidentes al 100 %, por lo que se debe llegar a un procedimiento aproximativo en el cual los puntos se acercan al máximo. El procedimiento entrega una matriz que se utiliza como inicio para, iterativamente, buscar una solución mejor dados los puntos vecinos de los originales. Aunque normalmente se aplica otro proceso de registrado para obtener un ajuste más fino (bastante costoso en tiempo), para nuestra aplicación no es necesario debido a la inherente falta de detalle que se observa en las imágenes generadas por punteado, en particular del ámbito de nuestro interés, monumentos, restos arqueológicos, etc.

Un aspecto importante que cabe destacar es la dificultad en la selección de los puntos que representan las triadas de las toma fija y flotante. Este proceso, en lo que conocemos, se realiza siempre de manera manual y basa su correcto funcionamiento en la capacidad del usuario para detectar puntos que representan posiciones iguales en ambos conjuntos de puntos. Dada la gran cantidad de puntos que se pueden obtener, este proceso se realiza teniendo en cuenta la geometría, pero sobre todo, la información de la textura, si se posee, para discriminar características visuales comunes. Esto no siempre es posible, por lo que dificulta gravemente el proceso.

En nuestro caso, hemos desarrollado una nueva alternativa basada en la siguiente idea: si el conjunto de vértices es más pequeño, la selección de puntos es más sencilla. Para ello utilizamos el algoritmo de Douglas-Peucker, al igual que lo hicimos para la visualización. Mediante este procedimiento, el usuario realiza una simplificación en ambas mallas, manteniendo los puntos representativos. Sobre ellos se hace la selección de las triadas. La reducción típica es del 90-95 % de los puntos.

El siguiente paso en los sistemas de tratamiento de datos escaneados consiste en la fusión de las mallas, esto es, crear un solo conjunto de puntos a partir de los puntos de cada toma y de las transformaciones de registrado. En nuestro caso esto no es necesario, ya que no buscamos otra capacidad del modelo más que la visualización y para ello no es necesario tener un solo conjunto de datos. Además, en algunos casos el solapamiento parcial de puntos permite añadir nuevos puntos a la visualización. En caso de que no sea así, es necesario eliminar los puntos que se solapan con otros de otra toma y que producen un resultado final no estético. Mediante una herramienta interactiva el usuario puede indicar que puntos serán los eliminados.

## 4.2. Añadiendo ruido para evitar patrones

Otro proceso importante para la generación de imágenes es la adición de ruido. Si visualizamos los puntos obtenidos directamente por el escáner, con o sin reducción, observaremos que la distribución de los puntos siguen un patrón preestablecido y visible: la captura de los datos por el escáner. El mismo hace una división regular del espacio de escaneado. Si la superficie escaneada es irregular en forma y/o existencia (huecos o falta de material) este patrón se ve disimulado, pero en otro caso es fácilmente discernible. Una de las características de las imágenes con punteado hecho a mano es que se evita a nivel macroscópico cualquier patrón o regularidad que pueda hacer que el observador fije su atención en el mismo. Por tanto, debemos eliminar dicha distribución regular. Para ello añadimos una herramienta que permite, de una forma interactiva, añadir el nivel de ruido deseado al conjunto de vértices seleccionado. Esto quiere decir que, por ejemplo, se puede añadir una mayor cantidad de ruido en zonas que sean especialmente importantes (paisaje, árboles, etc.) y permitir una menor cantidad en zonas donde se desee preservar la información original. Hay que observar que aunque este procedimiento de añadir ruido puede parecer que desvirtúa los resultados, por un lado, en el caso de la visualización, dicha exactitud no es tan necesaria, y por otro lado, cierto nivel de ruido es inevitable y se encuentra en los distintos procesos que se aplican: captura de datos, registrado y filtrado en la etapa de fusión. En esta última, los datos son combinados y para ello se suelen usar valores medios.

El proceso de añadir ruido se basa también en la estructura matricial en la que se encuentran los datos. El procedimiento consiste en mover un punto de forma aleatoria con respecto a sus vecinos. Para cada vecino, se define una recta, obteniendo una posición intermedia de forma aleatoria. Esta nueva posición se usa con el siguiente vecino de la misma forma y así sucesivamente. El resultado final es que el punto mantiene una posición coherente con su entorno pero rompiendo la regularidad del proceso de escaneado.

## 4.3. Generación de la imagen

En esta etapa se muestran las técnicas y métodos que se aplicarán para la obtención de las imágenes finales.

### 4.3.1. Simplificación

Una de las características del algoritmo de Douglas-Peucker es que, inherentemente, implementa un nivel de detalle [Cla76]. Dado que trabaja recursivamente sobre conjuntos de datos más pequeños y en los que va variando el umbral, el algoritmo cataloga los puntos dependiendo de una serie de valores. Si estos valores son usados como umbrales para eliminar o mantener puntos, estaremos hablando de un algoritmo que realiza un control en el nivel de detalle, desde el conjunto completo de puntos cuando el umbral es cero, hasta el conjunto mínimo, dos puntos, cuando el umbral es máximo.

Dada la implementación que se ha expuesto anteriormente del algoritmo de Douglas-Peucker, el usuario puede seleccionar de forma interactiva el conjunto de puntos que desea visualizar. Moviendo un deslizador entre cero y la distancia máxima calculada, permitirá mostrar desde la malla completa hasta un conjunto muy reducido de puntos. El procedimiento es muy rápido pues consiste en mostrar los puntos que estén por encima del umbral indicado.

Otra de las ventajas que aporta el Douglas-Peucker, por la forma en la que se ha implementado, es la de permitir una visualización dependiente de la distancia. Esto es, el conjunto de puntos mostrado varía dependiendo de la cercanía o lejanía del observador. Dado que el tamaño de los puntos es fijo, en principio 1 píxel, cuando el observador se acerca al objeto deben aparecer más puntos para cubrir una misma cantidad de espacio. Al contrario, cuando el observador se aleja, se deben eliminar puntos para evitar que se solapen unos con otros y aparezcan zonas manchadas. Se han implementados dos posibilidades. En la primera se establece una función de interpolación entre niveles umbrales de eliminación de puntos, según se ha explicado anteriormente. Si se dispone de un umbral 0 a la máxima cercanía, esto implica mostrar todos los puntos, y un umbral grande cuando el observador se aleja del objeto, lo cual supone mostrar menos puntos, se puede obtener una variación lineal en el valor del umbral que discrimina los puntos. La distancia que permite obtener el umbral se calcula a partir del centro de la caja frontera del objeto y la posición de la cámara. Hemos podido comprobar que aunque una función lineal da un resultado aceptable, es mucho mejor si se usa una función de un grado mayor. Para ello se ha implementado una función polinomial de grado 3, mediante un bspline, el cual permite una variación extremadamente correcta del conjunto de puntos en función de la distancia. La segunda posibilidad se basa en la misma idea de los umbrales en función de la distancia, pero lo aplica de una manera más fina, ya que la distancia se calcula entre la cámara y cada uno de los puntos. Aunque es un poco más lento, este procedimiento proporciona mejores resultados.

### 4.3.2. Iluminación

Una de las características de las imágenes con punteado es la posibilidad de mostrar el tono cambiando la densidad de puntos. Con el procedimiento descrito hasta ahora, usando el algoritmo de Douglas-Peucker, la densidad de puntos se cambia pero en base a condicionantes geométricos. Es posible simular la aplicación de una fuente luminosa utilizando la siguiente idea: deseamos que la densidad de puntos sea menor donde la iluminación sea más intensa y que la densidad de puntos sea mayor donde la iluminación sea menor. Para obtener este resultado necesitamos saber si un punto es iluminado y con qué intensidad y en función de dicho resultado hacerlo visible o no. Para poder calcular la reflexión de luz en un punto necesitamos la normal de dicho punto.

Dado que se han calculado, en una etapa previa, las normales de cada punto, se puede calcular la reflexión mediante

Modelo	Puntos	Caras	fps puntos	fps caras
Torre	151400	287201	195.72	82.42
Teatro	418128	793.369	153.53	38.61
Castillo	1519344	2766422	41.84	10.93

**Table 1:** Resultados.

el modelo sencillo de iluminación [FvDFH92]. La ecuación nos devuelve un valor que varía entre 0 y 1 para los puntos iluminados, siendo mayor que 0 el valor mínimo y 1 el valor máximo. Si el valor es menor que cero el punto no está iluminado. El resultado obtenido se utiliza como valor que cambia el umbral de selección del algoritmo de Douglas-Peucker. La implementación se basa en utilizarlo como un factor multiplicativo. Para controlar la iluminación el usuario puede mover las fuentes de luz a las posiciones que considere apropiadas para obtener el resultado deseado.

## 5. Resultados

A continuación se comentan los resultados obtenidos.

La Figura 3 muestra la diferencia de operar con el conjunto completo de puntos y sólo con los más importantes. La Figura 4 representa el resultado final. La regularidad del escaneado se elimina mediante la adición de ruido, como se muestra en la Figura 5. La Figura 6 muestra cómo afecta a la imagen la variación del umbral que utiliza el algoritmo de Douglas-Peucker. En la Figura 7 se puede ver el efecto de la distancia sobre el conjunto de puntos. Obsérvese cómo en todas las imágenes, el modelo es perfectamente visible. Ésto sería imposible, sobre todo en las imágenes con la cámara más alejada, pues se dibujarían todos los puntos en una área muy pequeña. El efecto de aplicar la iluminación se muestra en las imágenes de la Figura 8. Las imágenes muestran una fuente de luz que gira alrededor del objeto. En la Figura 9 se muestran otros ejemplos de monumentos.

Se ha usado la siguiente configuración: Intel Core 2 Duo 6600, Nvidia 6800 GS (256MB), 2GB de RAM, S.O. Linux 2.6.20 y compilador gcc 4.1.1. Los resultados numéricos se muestran en la Tabla 1 (téngase en cuenta la antigüedad de la tarjeta gráfica para valorar las cifras).

## 6. Conclusiones

Se ha presentado un método para la creación de imágenes con la técnica del punteado a partir de datos escaneados de monumentos y restos arqueológicos. Habiendo otras soluciones que buscan el mismo objetivo, el trabajo presentado aporta las siguientes ventajas y aportaciones:

- Se ha desarrollado un nuevo procedimiento que facilita el registro. Frente a los métodos normales, la simplificación previa de la malla facilita mucho el proceso de selección de los puntos.

- Es intuitivo y fácil de usar por el usuario. En tiempo real el usuario cambia el conjunto de puntos que se visualizan con un simple deslizador. Moviendo las luces controla el tono.
- Los resultados son muy buenos. Los comentarios de la persona que ha colaborado indica que en muchos casos el resultado es bastante parecido al creado por un humano.
- Es rápido, trabajando en tiempo real con cientos de miles de puntos. El pre-procesamiento de los triángulos, normales y distancias del algoritmo de Douglas-Peucker hacen que la visualización sea un proceso muy simple.
- Implementa un método de nivel de detalle. El algoritmo de Douglas-Peucker no sólo permite controlar los puntos que se muestran dependiendo de los gustos del usuario sino también en función de la distancia.
- Permite la animación con coherencia entre fotogramas. Al trabajar con los puntos originales de la malla y aplicar un procedimiento que realiza una selección de los puntos de forma continua (teniendo en cuenta que un punto sólo puede ser dibujado o no), ha permitido la generación de animaciones con coherencia entre fotogramas.

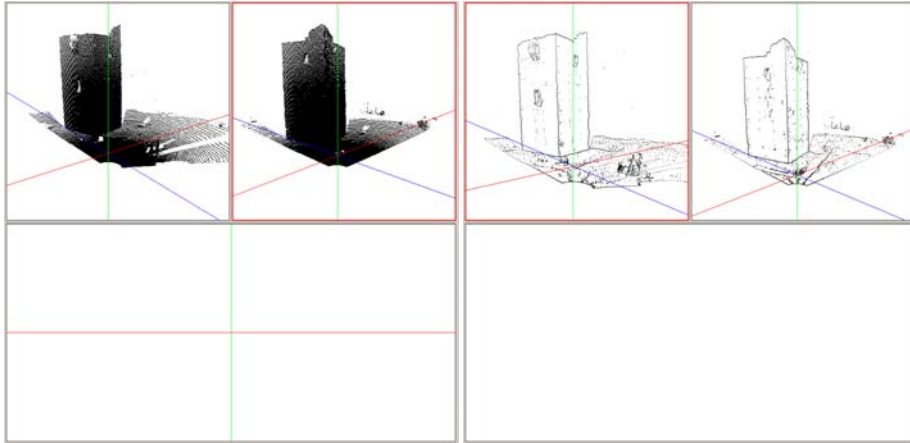
## Agradecimientos

Los autores desean agradecer la colaboración prestada por la ilustradora Elena Piñar Toro. Agradecimientos al Ministerio de Educación y Ciencia, por los proyectos TIN2007-67474-C03-02, TIN2007-67474-C03-01 y al programa José Castillejo de jóvenes doctores con referencia JC2007-00032, que han financiado este trabajo. También agradecemos a los revisores sus comentarios y sugerencias para mejorar este artículo.

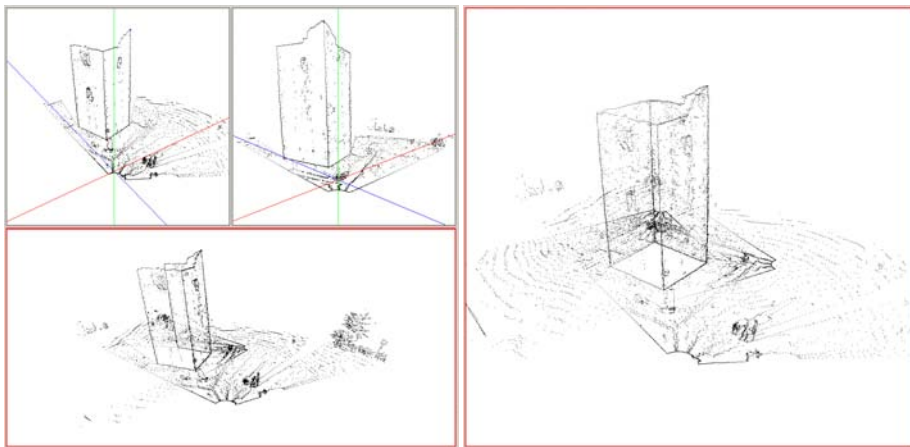
## References

- [Cla76] CLARK J. A.: Hierarchical geometric models for visible-surface algorithms. *Communications of the ACM* 19, 10 (October 1976), 547–554.
- [DHvOS00] DEUSSEN O., HILLER S., VAN OVERVELD C., STROTHOTTE T.: Floating points: A method for computing stipple drawings. *Computer Graphics Forum* 10, 3 (September 2000), 41–50.
- [DP73] DOUGLAS D. H., PEUCKER T. K.: Algorithms for reductions of the number of points required to represent a digitised line or its caricature. *The Canadian Cartographer* 2, 10 (September 1973), 112–122.
- [DS00] DEUSSEN O., STROTHOTTE T.: Computer-generated pen-and-ink illustration of trees. *Proceedings of SIGGRAPH* (July 2000), 13–18.
- [FvDFH92] FOLEY J., VAN DAM A., FEINER S., HUGHES J. F.: *Computer Graphics: Principles And Practice, 2 Edition*. Addison-Wesley, 1992.

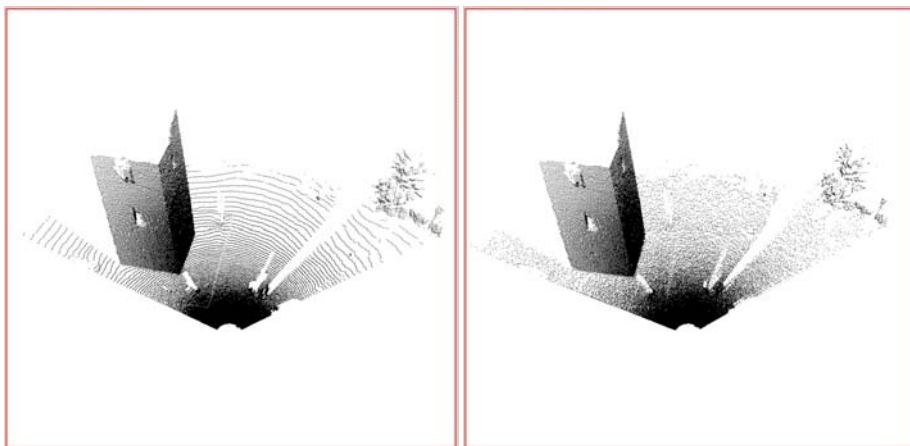
- [GG01] GOOCH B., GOOCH A.: *Non-Photorealistic Rendering*. A.K. Peters, 2001.
- [Hod89] HODGES E.: *The Guild Handbook of Scientific Illustration*. John Wiley and Sons, 1989.
- [Hop96] HOPPE H.: Progressive meshes. *Proceedings of SIGGRAPH* (August 1996), 99–108.
- [Hor87] HORN B. K. P.: Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A* 4 (April 1987), 629–642.
- [LM\*03] LU A., MORRIS C. J., TAYLOR J., EBERT D. S., HANSEN C., RHEINGANS P., HARTNER M.: Illustrative interactive stipple rendering. *IEEE Transactions on Visualization and Computer Graphics* (2003), 127–138.
- [LME\*02] LU A., MORRIS C. J., EBERT D. S., RHEINGANS P., HANSEN C.: Non-photorealistic volume rendering using stippling techniques. *Proceedings of Visualization'02* (2002), 211–218.
- [LS95] LANSDOWN J., SCHOFIELD S.: Expressive rendering: A review of non-photorealistic techniques. *IEEE Computer Graphics and Applications* (May 1995), 29–37.
- [LW85] LEVOY M., WHITTED T.: *The use of points as a display primitive*. Tech. rep., University of North Carolina-Chapel Hill Computer Science Department, 1985.
- [M\*97] MARKOSIAN L., ET AL.: Real-time non-photorealistic rendering. *Proceedings of SIGGRAPH* (1997), 415–419.
- [M\*00] MARKOSIAN L., ET AL.: Art-based rendering with continuous levels of detail. *Proceedings of NPAR 2000* (June 2000), 59–66.
- [MIA\*08] MACIEJEWSKI R., ISENBERG T., ANDREWS W. M., EBERT D. S., SOUSA M. C., CHEN W.: Measuring stipple aesthetics in hand-drawn and computer-generated images. *IEEE Computer Graphics and Applications* (2008), 62–74.
- [PFS03] PASTOR O. M., FREUDENBERG B., STROTHOTTE T.: Real-time animated stippling. *IEEE Computer Graphics and Applications* 23, 4 (2003), 62–68.
- [Sec02] SECORD A.: Weighted voronoi stippling. *Proceedings of NPAR'02* (2002), 37–43.
- [SHS02] SECORD A., HEIDRICH W., STREIT L.: Fast primitive distribution for illustration. *Proceedings of the 13th Eurographics workshop on Rendering 28* (2002), 215–226.
- [SS02] STROTHOTTE T., SCHLECHTWEIG S.: *Non-Photorealistic Computer Graphics: Modeling, Rendering and Animation*. Morgan Kaufmann, 2002.
- [Woo94] WOOD P.: *Scientific illustration, second edition*. Vand Nostrand Reinhold, 1994.
- [WS94] WINKENBACH G., SALESIN D. H.: Computer generated pen and ink illustration. *Proceedings of SIGGRAPH* (July 1994), 469–476.
- [WS96] WINKENBACH G., SALESIN D. H.: Rendering parametric surfaces in pen and ink. *Proceedings of SIGGRAPH* (August 1996), 469–476.
- [XC04] XU H., CHEN B.: Stylized rendering of 3d scanned world environments. *Proceedings of NPAR'04* (2004), 25–34.
- [YNZC05] YUAN X., NGUYEN M. X., ZHANG N., CHEN B.: Stippling and silhouettes rendering in geometry-image space. *Proceedings of the Eurographics Symposium on rendering'05* (2005), 173–176.
- [ZF\*01] ZWICKER M., PFISTER H., VAN BAAR J., GROSS M.: Surface plating. *Proceedings of SIGGRAPH* (August 2001), 371–378.
- [ZS04] ZAKARIA N., SEIDEL H. P.: Interactive stylized silhouette for point-sampled geometry. *Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia* (2004), 242–249.



**Figure 3:** Registrado usando la malla completa o sólo los puntos más importantes.

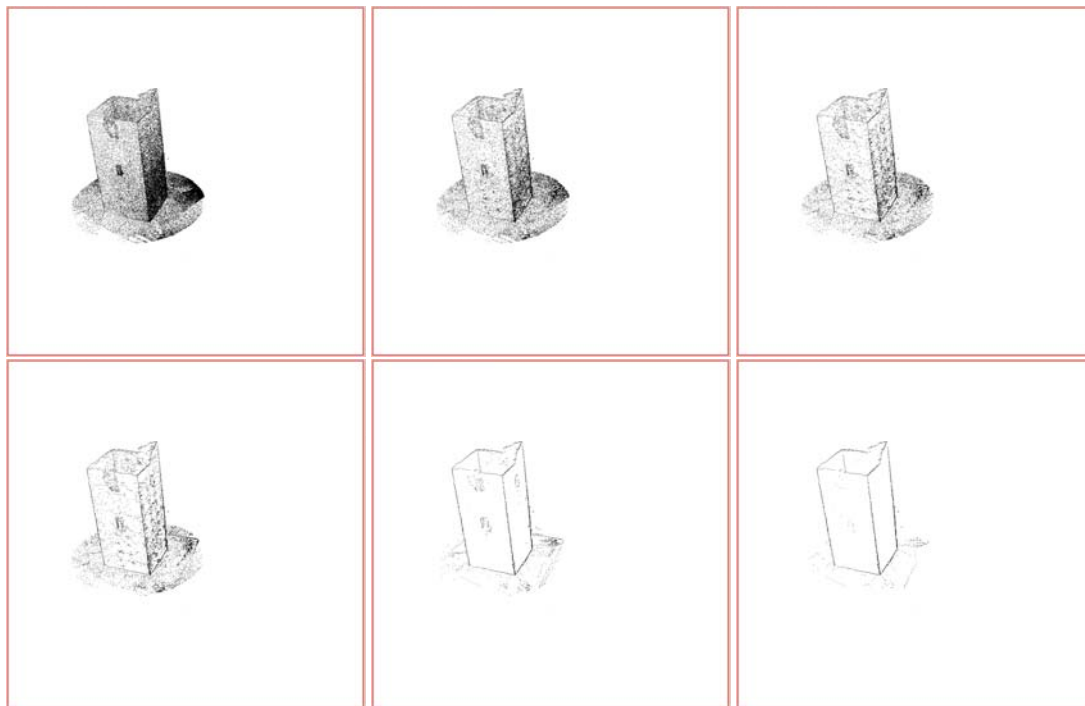


**Figure 4:** Resultado del registrado.

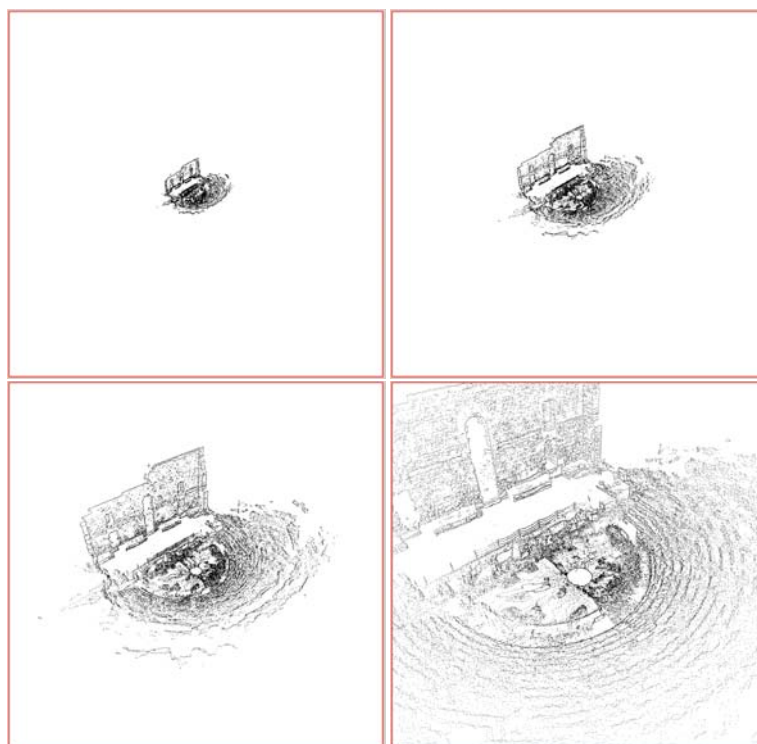


**Figure 5:** Reduciendo el ruido.

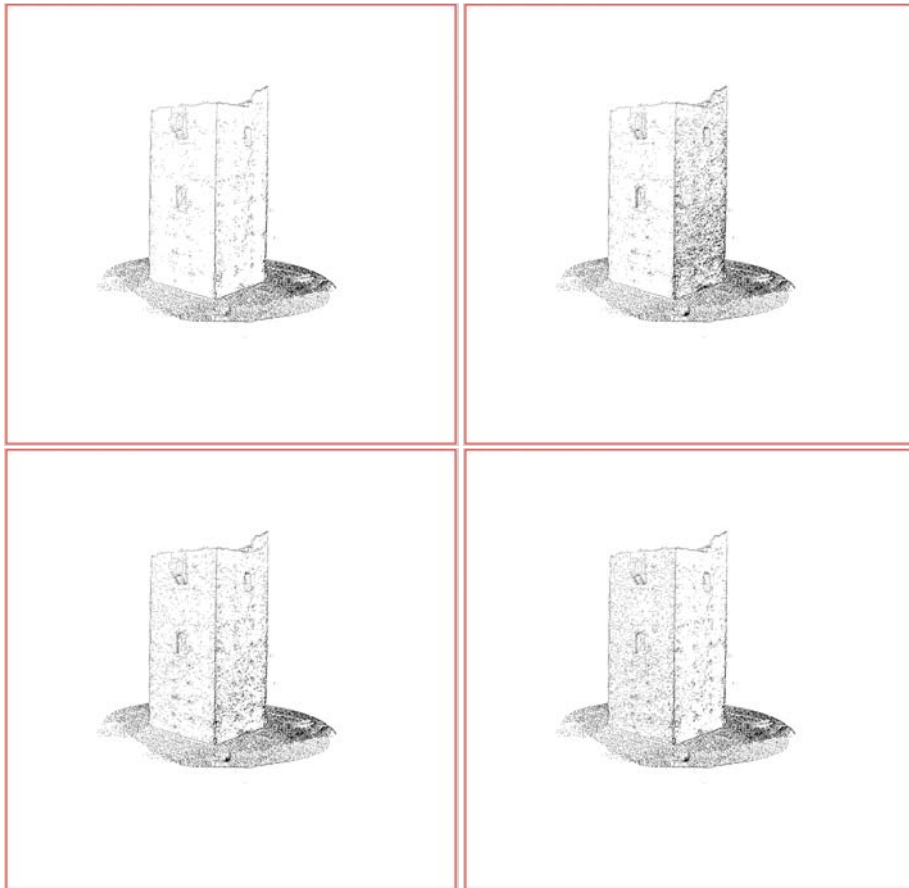




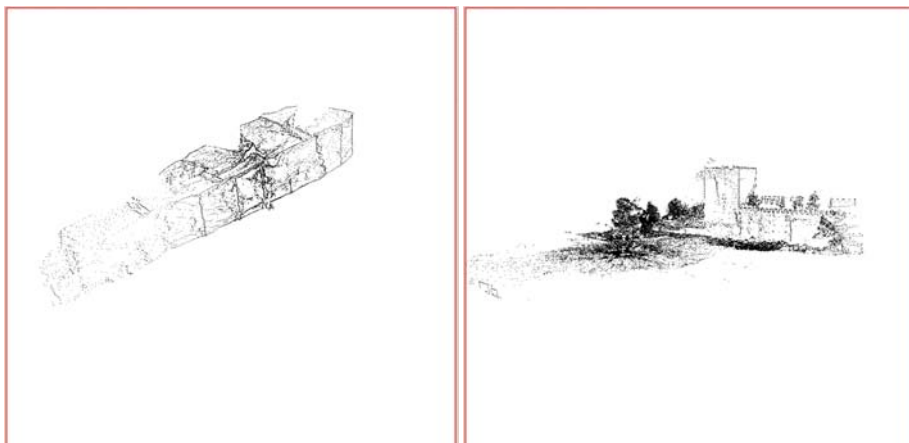
**Figure 6:** Variando el umbral.



**Figure 7:** Controlando el número de puntos en función de la distancia.



**Figure 8:** *Añadiendo la iluminación.*



(a) Dolmen megalítico

(b) Castillo

**Figure 9:** *Otros modelos.*