
DISEÑO E IMPLEMENTACIÓN DE UNA BASE DE DATOS DE IMÁGENES DE RANGO CON DECIMACIÓN PARA LA VISUALIZACIÓN EN WEB

•
Isabel Rodríguez*¹
Alexánder Ceballos*²
Jorge Hernández*³
Flavio Prieto*⁴

•

Recibido: 10/02/2007

Aceptado: 28/05/2007

Resumen

Las bases de datos son una herramienta para administrar y almacenar la información, mientras mejor esté distribuida, más fácil y eficiente será el acceso a esta. En la actualidad, y gracias al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos tienen formato electrónico, ya que ofrece un amplio rango de soluciones al problema de almacenar datos.

En este artículo, presentamos el diseño de una base de datos para imágenes de rango, basada en el modelo relacional y una metodología de diseño consistente en tres pasos: diseño conceptual, diseño lógico y diseño físico. Dicha base de datos cuenta con acceso vía Web. Un algoritmo de multirresolución fue desarrollado e implementado para mejorar el costo computacional en visualización y renderización, utilizando parámetros de decimación basados en curvatura.

Se desarrolló una plataforma Web para el acceso de forma amigable a la base de datos para que se puedan ingresar, actualizar y eliminar modelos, de forma confiable y sencilla. Actualmente, la base de datos cuenta con 40 modelos divididos en 3 categorías: rostros (22), museo (4) y objetos de forma libre (14). El algoritmo de multirresolución usado, arrojó resultados de ahorro en costo computacional hasta del 96%.

Palabras clave

Bases de datos, imágenes de rango, multirresolución, decimación.

* Departamento de Ingeniería Eléctrica, Electrónica y Computación Universidad Nacional de Colombia, sede Manizales

¹ mrodriguez@unal.edu.co

² aceballosa@unal.edu.co

³ jehernandez@unal.edu.co

⁴ faprieto@unal.edu.co

DESIGN AND IMPLEMENTATION OF A RANGE IMAGES DATABASE WITH DECIMATION TO BE VISUALIZED IN THE WEB

Abstract

Databases are tools to store and manage information, the better they are distributed, the easier and more efficient will its access be. At present, thanks to technological development of fields such as computer science and electronics, most of them have an electronic format, since they offer a wide rank of solutions to the store data problem.

In this paper, we present the design and development of a database for range images, based on the relational model and a design methodology consisting of three steps: conceptual design, logical design, and physical design. The data base has a WEB access. A multi-resolution algorithm was developed to improve the time cost in visualization and renderization, by using decimation parameters based on the curvature.

A WEB interface was developed in order to provide an easy access, consequently, it is possible to enter, update, and eliminate models on a reliable and simple way. At present, the database counts on 40 models divided in 3 categories: Faces (22), Museum (4) and objects of free form (14). The use of the multi-resolution algorithm allows saving up to 96% of computational cost.

Key Words

Database, range images, multi-resolution, decimation.

INTRODUCCIÓN

Una base de datos es un conjunto de datos que pertenecen al mismo contexto, almacenados sistemáticamente para su uso posterior. Debido a la adquisición de una cámara de rango (Vivid 9i) por parte de la Universidad Nacional de Colombia, sede Manizales, se generó la necesidad de poseer una base de datos con la información de los modelos tridimensionales obtenidos con dicha cámara, para tener una organización adecuada de los modelos y tener acceso a ellos mediante una aplicación WEB.

Las bases de datos relacionales son el método de diseño más difundido actualmente entre los creadores de software. Los motivos de este éxito son fundamentalmente dos: *i)* Ofrecen sistemas simples y eficaces para representar y manipular los datos, *ii)* Se basan en el modelo relacional, propuesto originariamente por Edgar Frank Codd (Codd, 1970). Gracias a su coherencia y facilidad de uso, el modelo se ha convertido desde los años 80 en el más usado para la producción de bases de datos. Por tanto, es el modelo por excelencia para crear una BD eficiente y fácil de usar, donde se almacenan datos de diferentes tipos y se organizan en tablas, de tal forma que se lean los atributos en las columnas, y las características o registros individuales de dichos atributos, en las filas.

Las imágenes de rango constituyen una manera no invasiva (que no requiere contacto con el objeto), y bastante aceptable de obtener información acerca de la geometría de un objeto. Son “fotografías digitales” del objeto, donde en vez de capturar valores de color, intensidad, luminosidad, etc., se conforma un arreglo de distancias tomadas desde el sensor hasta la superficie del objeto (Gómez, 2004). Las imágenes de rango pueden ser representadas de dos formas básicas: *i)* Por una lista de coordenadas 3D en un marco de referencia dado (nube de puntos), para lo cual no se requiere ningún orden específico. *ii)* Por una matriz

de valores de profundidad de puntos a lo largo de los ejes X y Y de la imagen, lo cual hace explícita la organización espacial (Catnzler, 1997).

Dado el gran número de puntos o coordenadas, se pueden simplificar imágenes de rango representándolas con un menor número de polígonos y de puntos mediante un proceso llamado decimación, lo que permite que el costo computacional, de almacenamiento y de renderización disminuya considerablemente. Se comienza con una superficie muy fina o sobremuestreada y se remueven primitivas como triángulos, vértices, bordes, huecos, túneles o cavidades.

El escáner VIVID 9i es un digitalizador 3D de no-contacto. Dicho dispositivo es ideal para ingeniería inversa, verificación de diseño, inspección de calidad y otras aplicaciones industriales. Además de capturar la imagen de rango, el escáner brinda información de textura en formato de mapa de bits, con una resolución de 640x480 en un tiempo de escaneo de 2.5 segundos. El formato original de los archivos es VVD, aunque pueden ser guardados en otros formatos como PLY y WRL.

En el marco de BD de imágenes de rango, una de las más conocidas en el medio es el OSU - Base de Datos de Imágenes de Rango del *Signal Analysis and Machine Perception Laboratory*, del Departamento de Ingeniería Eléctrica de la Universidad de Ohio. Esta base de datos contiene imágenes escaneadas con varios sensores de rango reales y en su mayoría con datos sintéticos, disponibles en los formatos GIF y formato neutral comprimido (formato '.txt') con dimensiones de puntos en X, Y y Z. El Repositorio de *University of Stanford* posee imágenes de rango tomadas con varios sensores, en formato PLY, y que contiene imágenes del proyecto Miguel Ángel. Otra BD es la de la USF (*University of South Florida*) la cual contiene alrededor de 400 imágenes de rango, tomadas desde diferentes cámaras, como Odetics LADAR, Percetron LADAR, de Luz estructurada ABW y

de luz estructurada K2T y, por consiguiente, dadas en diferentes formatos como PGM, ABW y K2t entre otros.

Las BD anteriores, algunas más robustas que otras, son ricas en información para la digitalización del objeto, pero en esta revisión, no se ha observado alguna que contenga un algoritmo de decimación de las imágenes de rango para una visualización más rápida en su acceso vía WEB, siendo esto último uno de los principales aportes presentados en este trabajo.

Este artículo está organizado de la siguiente manera: en la primera sección se muestra el desarrollo del diseño e implementación de la base de datos; en la segunda sección se presenta el desarrollo del algoritmo de multirresolución para la visualización de los modelos en la plataforma WEB; en la tercera sección se discuten los resultados de la base de datos y el ahorro en costo computacional después de aplicar un algoritmo de multirresolución, para los modelos relacionados en la BD. Finalmente, se presentan las conclusiones.

DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS

En la determinación de las fases de la metodología para la creación de una base de datos, se debe definir una jerarquía de niveles que resulte apropiada en el sentido de ser lo suficientemente amplia para que a cada nivel le correspondan decisiones de diseño bien definidas, pero, a la vez, no proponer demasiados niveles, ya que sería muy sensible a la interpretación del diseñador. Durante el proceso, se distinguieron las siguientes etapas para el diseño de la base de datos:

- Diseño conceptual
- Diseño lógico
- Diseño físico

Diseño conceptual

Esta etapa recibe como entrada la especificación de requerimientos, y su resultado es el esquema conceptual de la base de datos, que es una descripción de alto nivel de la estructura de ella, independiente del software que se use para manipularla. La BD UN-MZLS-MODELOS tiene como objetivo la unión de las diferentes características de los modelos obtenidos mediante el escáner de rango, de tal forma que se obtenga organizadamente todos los valores referentes al modelo y cada una de sus vistas. Además, contempla una lista de usuarios cuyas funciones dependen de los privilegios asignados entre ellos: ingresar, actualizar y borrar modelos. Para esto utilizamos el modelo conceptual entidad-relación, cuyo paso inicial es el análisis de requisitos. Se definieron parámetros específicos para cada modelo, cada vista y cada usuario.

1. Entidad modelos

- ID: identificador de cada modelo, primordial para la selección de modelos.
- Código de modelo: Valor único, creado para reconocer cada modelo sin confundirlo con los demás, y de esta forma conocer los valores únicos de sus atributos.
- Nombre del modelo: Con este nombre se identifica el modelo físico que se obtuvo mediante el escáner de rango.
- Número de puntos: Para cada modelo se especifica un único número total de puntos, el cual brinda una idea global de la nube de puntos del modelo.
- Número de triángulos: Número único de triángulos que conforman el modelo; dicho valor se hace útil en diferentes algoritmos de tratamiento de modelos 3D.
- Número de vistas: Número de adquisiciones realizadas al modelo, necesarias para su posterior registro; dicho valor es útil para la crea-

ción de la base de datos de las vistas. Un modelo puede tener una o varias vistas.

- **Categoría del modelo:** Los modelos se dividen en 3 categorías: museo, rostros y otros; con este atributo se distingue entre ellos y se reconoce la carpeta donde debe guardarse, sin embargo, el número de categorías se puede ampliar.
- **Fecha de registro:** Fecha en la cual se realizó el registro del modelo con las diferentes vistas; no necesariamente es igual a la fecha de las vistas.
- **Textura:** Parámetro para reconocer si el modelo es texturado o no. Por textura, nos referimos a las imágenes de intensidad.
- **Ruta de modelo:** Directorio en el cual se encuentra el modelo registrado y decimado en formato wrl para la visualización en la implementación WEB de la base de datos.
- **Descripción del modelo:** resumen de las características más relevantes del modelo, entre ellas, el material, su naturaleza y procedencia.
- **Posición de la cámara:** Valores XYZ de las coordenadas donde se encuentra el escáner de rango, con el cual se obtuvo el modelo.

En la figura 1, presentamos un ejemplo. Se puede observar el modelo del rostro 1, con sus respectivas vistas, mientras en la tabla 1, se encuentran las características del modelo, almacenadas en la base de datos.



(a) Rostro 1 completo



(b) Rostro 1 nube de puntos



(c) Rostro 1 triangulado

Figura 1. Visualización de modelos

Tabla 1: Consulta en base de datos para un modelo

Código:	RISA080306
Nombre:	Rostro_isabel
Categoría:	Rostros
Fecha de registro:	2006-03-08
Número de vistas:	1
Número de puntos:	113328
Número de triángulos:	222463
Descripción:	Rostro, intensidad baja para evitar el color negro de la imagen, toma única a 0°.
Texturado:	si
Posición XYZ:	-16.591602 -13.097415 -445.007059

Entidad vistas

- ID de vista: Identificador de cada vista y llave principal de la entidad.
- Código de vista: Valor único, creado para reconocer cada vista.
- Nombre de vista: Con este nombre se identifica el modelo físico y el número de vista que corresponde al modelo que se obtuvo mediante el escáner de rango.
- Número de puntos: Para cada vista se especifica un único número total de puntos, el cual brinda una idea global de la nube de puntos de la vista.
- Número de triángulos: Número único de triángulos que conforman la vista.
- Fecha de vistas: Fecha en la cual se realizaron las vistas del modelo.
- Textura: Atributo para reconocer si la vista tiene textura o no.
- Ruta de vista: Directorio en el cual se encuentra la vista; al realizar la implementación WEB de la base de datos con PHP, se utilizará este campo para la visualización y las descargas.
- Descripción de la vista: Explica si la textura fue tomada con el escáner de rango, con la cámara Sony o con ambas; esto con el fin de conocer la dimensión de la textura.
- Ángulo de la vista: Valor del ángulo de la tornamesa con respecto al escáner; éste valor es importante para un mejor registro de las vistas.

Un ejemplo es presentado en la figura 2, en esta se puede observar la vista del modelo del ejemplo presentado en la figura 1, mientras en la tabla 2, se encuentran las características del modelo, almacenadas en la base de datos.

- (a) Vista 1 completa
- (b) Vista 1 nube de puntos
- (c) Vista 1 triangulada



a) Vista 1 completa



(b) Vista 1 nube de puntos



(c) Vista 1 triangulada

Figura 2. Visualización de vista

Tabla 2. Consulta en base de datos para una vista

Código:	CISADATA0
Nombre:	rostro2_toma1
Modelo:	Rostro_isabel
Fecha de registro:	2006-03-08
Número de puntos:	225128
Número de triángulos:	116010
Descripción:	Toma sin textura, recorte de la toma rostro Isabel.
Texturado:	No.
Angulo:	0°

- IDU: Identificador de usuario, dicho valor debe ser único siendo el administrador el IDU: 1.
- Nombre de usuario: Nombre del usuario que será utilizado al ingresar a la página para reconocer si tiene o no privilegios.
- Contraseña: Valor que será ingresado para reconocer el usuario y que tendrá directa relación con el nombre de usuario.
- Correo electrónico: Dirección para enviar algún comentario.
- Tipo de usuario: Campo para conocer los privilegios del usuario, puede ser: admin, privado u otro.

ENTIDAD USUARIOS

Los usuarios que se permiten son de tres tipos: administrador, usuario para descargar y privilegiado. El administrador será una única persona con diferentes privilegios: ingresar modelos, modificarlos, ingresar y actualizar los datos de los usuarios privilegiados. Los usuarios para descargar podrán descargar los modelos completos en los formatos almacenados en la base de datos. Finalmente, los usuarios privilegiados también tienen permiso de ingresar nuevos modelos. Las características de la entidad de usuarios son las siguientes:

Teniendo en cuenta cada uno de los requisitos especificados anteriormente, se realiza el modelo conceptual. Se observa en la figura 3, que existe una relación 1:n entre la entidad 'Modelo' y la entidad 'Vista'. Por tanto, por cada modelo existirán una o varias vistas, que serán almacenadas en la tabla vistas de la base de datos y la relación se establece a través del nombre del modelo, es decir, se agrega a esta entidad un campo con el nombre del modelo que corresponde a la vista.

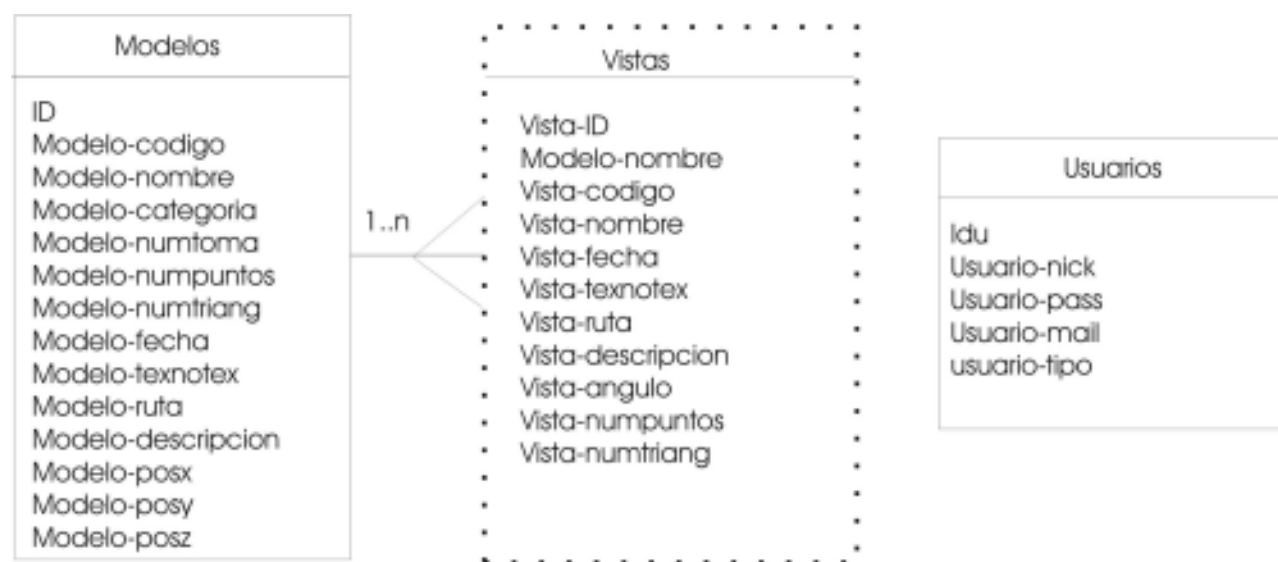


Figura 3: Modelo entidad relación

Diseño lógico

En esta etapa se transforma el esquema conceptual obtenido en la fase anterior a un esquema relacional. Para este caso, todos los datos del diseño conceptual están estructurados en el nivel lógico, como tablas formadas por filas y columnas. Dado que en el modelo conceptual se crearon tres entidades: Modelos, Usuarios y Vistas, en esta fase se convertirán en tablas, poniendo en las columnas los atributos, y en las filas las características que corresponden a cada campo. Cada propiedad tiene un tipo de variable diferente como se explica a continuación:

- ID, IDU, vista_ID: tipo entero, índice único autoincrementado.
- Código de modelo, código de vista: tipo carácter, índice, llave primaria única.
- Nombre del modelo, nombre de vista, nombre de usuario: tipo carácter.
- Número de puntos, número de triángulos, número de vistas: tipo entero.
- Fecha de vistas, Fecha de registro: tipo date.
- Modelo Texturado/No Texturado, vista texturada/no texturada: tipo char (sí, no).
- Ruta de modelo, ruta de vista: tipo text.
- Descripción del modelo, Descripción de vista: tipo texto largo.
- Posición de la cámara: tipo doble para cada coordenada.
- Ángulo de vista: tipo entero.
- Contraseña de usuario: tipo carácter.
- Correo electrónico, tipo de usuario: tipo texto.

Diseño físico

El diseño físico recibe como entrada el esquema lógico y da como resultado un esquema físico, que

es una descripción de la implementación de una base de datos, describe las estructuras de almacenamiento y los métodos usados para tener un acceso efectivo a los datos. A continuación, se encuentran los pasos a seguir para la creación de la base de datos en un sistema de gestión de bases de datos (SGBD) como MySQL.

1. Crear la base de datos UN_MZLS_MODELOS.
2. Crear las tablas 'Modelos', 'Vistas' y 'Usuarios' con los campos descritos en el esquema lógico.
3. Llenar la tabla 'Modelos' con los valores insertados en modelosbd.txt, la tabla 'Vistas' con el archivo vistasbd.txt.
4. La tabla usuarios se llena directamente en la base de datos.

MULTIRRESOLUCIÓN PARA VISUALIZACIÓN

Uno de los mayores desafíos en visión artificial es conservar la calidad de las escenas reduciendo el costo computacional (Chacón, 2000; Kim). En imágenes 3D, esto puede lograrse al reducir el número de polígonos que componen los objetos para aquellos ubicados lejos del punto de vista del observador. También, para objetos adquiridos con diferente hardware, el sobremuestreo puede hacer que exista información innecesaria haciendo que el modelo sea computacionalmente pesado. Estos modelos se pueden simplificar sin que el ojo humano distinga la diferencia.

En la simplificación poligonal o decimación, la idea es reducir el número de triángulos, vértices, bordes, huecos, túneles o cavidades, agrupando puntos y haciendo necesaria, casi siempre, una nueva triangulación (Chacón, 2000, Kalvin, 1996, Rossignac, 1993). Existe una aproximación muy popular en la cual no es necesaria una nueva triangulación, llamada colapso de bordes, la

idea es remover puntos pertenecientes al mismo triángulo y conservar la triangulación de los que los rodean.

Para la visualización de los modelos en la plataforma Web, se pretende hacer uso de la desviación normal promedio como característica para realizar la simplificación de la malla en las regiones que brindan menos información (regiones planas), tratando de conservar aquellas más importantes en donde la forma local varía (las que presentan mayor curvatura).

Método adaptativo de multirresolución

Para identificar los puntos de la malla triangular que ofrecen poca información, se obtuvo una medida de la curvatura local de la superficie. Una buena aproximación es la desviación normal promedio, para lo cual se requiere la extracción de los vectores normales a la superficie en cada vértice del polígono (Castrillón, 2002, Papaioannou, 2000).

En un objeto 3D se puede definir una región superficial como una colección de polígonos conectados, cuyas normales tienen orientación similar. θ es el ángulo entre el vector normal al punto y el vector normal a la región. El vector normal promedio de una región puede ser visto como la suma de todos los vectores normales de los polígonos que la forman, ponderados por el área de cada uno (ecuación 1).

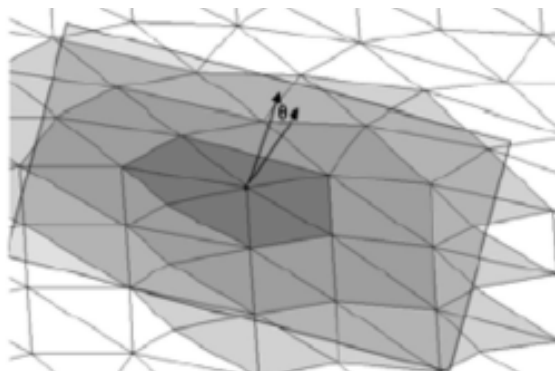


Figura 4: Orientación del vector normal al punto y del vector normal a la región.

$$\vec{N}_{prom}(R_k) = \left[\sum_{j|P_j \in R_k} A_j \vec{N}_j \right]^{-1} \cdot \sum_{j|P_j \in R_k} A_j \vec{N}_j \quad (1)$$

Donde A_j es el área del polígono j , perteneciente a la región k .

Para el cálculo del ángulo entre el vector normal a un punto y el vector normal a la región, se puede usar el producto punto entre los vectores N_i y $N_{prom}(R_k)$. (Ecuación 2).

$$\vec{N}_i \cdot \vec{N}_{prom}(R_k) = \cos(\theta) \quad (2)$$

Donde θ es la desviación normal promedio para el polígono i , perteneciente a la región k , y por lo tanto:

$$\theta = \cos^{-1}(\vec{N}_i \cdot \vec{N}_{prom}(R_k)) \quad (3)$$

Debe determinarse la región R_k sobre la cual se realizará el análisis, una buena aproximación puede ser hecha al considerar una región que rodee al punto. Una vecindad “sombrilla” es una región compuesta por los triángulos que rodean a un punto (figura 5); para vecindades más grandes se puede expandir la definición al incluir aquellos triángulos que rodean la primera región.

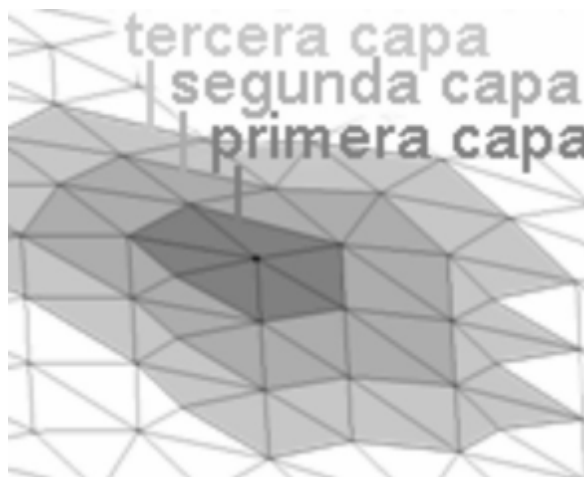


Figura 5: Tercera capa de la región sombrilla

Para la simplificación de la malla el método implementado fue el colapso de bordes, ubicando el nuevo vértice en la mitad del triángulo (Gieng, 1997). Los tres vértices de un triángulo son unidos en uno, ubicado en el promedio de

sus posiciones (figura 6). Con esto, desaparece el triángulo y también los tres triángulos adyacentes a sus lados, por lo tanto, el número total de puntos decrece en dos, mientras el de triángulos lo hace en cuatro.

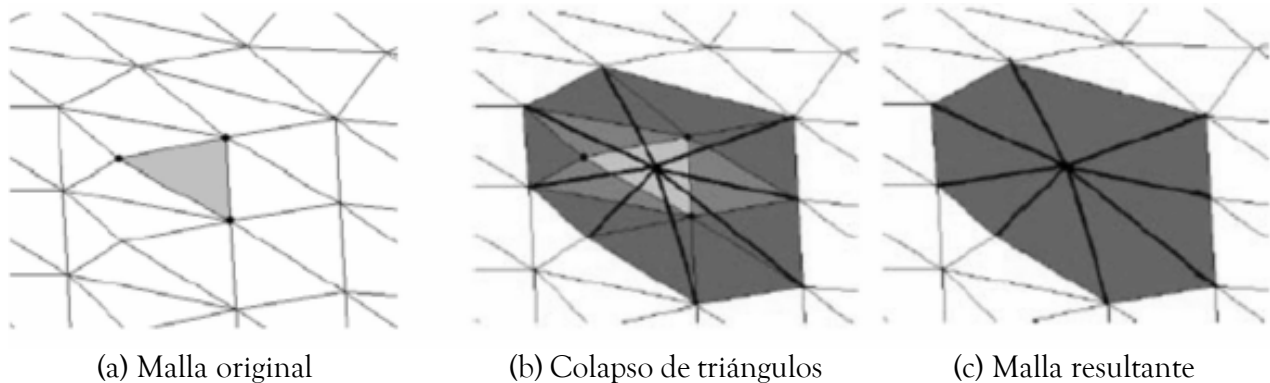


Figura 6: Colapso de triángulos

Algoritmo

Partiendo del modelo triangulado, se pueden hallar los vectores normales a cada triángulo. Identificar los bordes para que no sean removidos. Calcular la desviación normal promedio para cada punto determinado de la región sombrilla, en este caso, hasta la tercera capa. Finalmente se debe simplificar el modelo (algoritmo 1).

Algoritmo 1 Multirresolución(P, T)

Entradas: Conjunto de puntos $P_1, P_2, \dots, P_n \in P$.

Triangulación T de P .

Salida: puntos P y triángulos T pertenecientes al nuevo modelo simplificado.

Un triángulo t tiene 3 vértices, $t = (a, b, c)$;

Paso 1: Se identifican los puntos P_i que conforman el borde

Paso 2: Se calculan todas las normales de los triángulos t_i

Paso 3: Se asigna como vector normal a cada punto p_j el promedio de las normales de los triángulos a los cuales pertenece.

Paso 4: Se determina la tercera capa de la región sombrilla a cada punto P_j agregando en una lista *Vecindad_j* los triángulos t_i a los que éste pertenece y haciendo lo mismo con los primeros y segundos puntos vecinos.

Paso 5: Se calcula el vector normal a la vecindad sombrilla de cada punto P_j

Paso 6: Se calcula la desviación normal promedio para cada punto P_j como el ángulo entre su vector normal y el de su vecindad *Vecindad_j*.

Paso 7: Decimación: se colapsa el triángulo t_j si sus tres vértices poseen curvatura baja y no son bordes, además, si ninguno de sus triángulos vecinos ya ha sido colapsado.

A continuación presentamos los resultados obtenidos sobre una imagen de rango de un rostro que posee 152970 triángulos y una de una

escultura precolombina de 485340 triángulos. Se puede ver en la figura 7, que en la tercera iteración con el 11% de los triángulos originales y en la cuarta con el 6% la calidad visual sigue siendo buena para el caso del rostro, mientras que para la escultura la calidad es buena

aún en la tercera y cuarta iteración con el 8% y el 3% (figura 8). La malla se degrada de forma regular al mantenerse en proporción el ancho y alto de los triángulos, lo que hace que se logre un buen grado de decimación con buenos resultados.

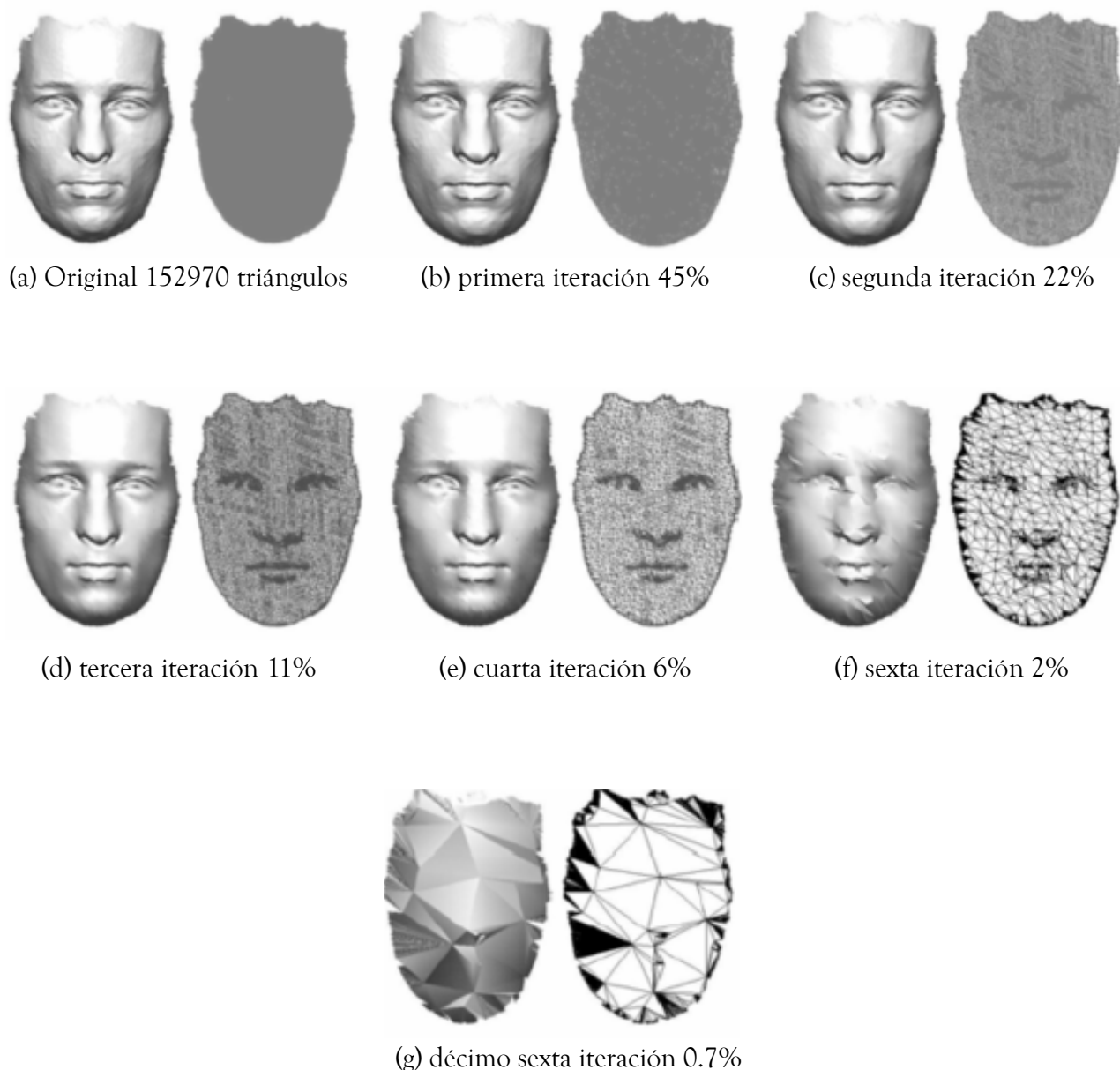


Figura 7: Colapso de triángulos realizado sobre el modelo de un rostro

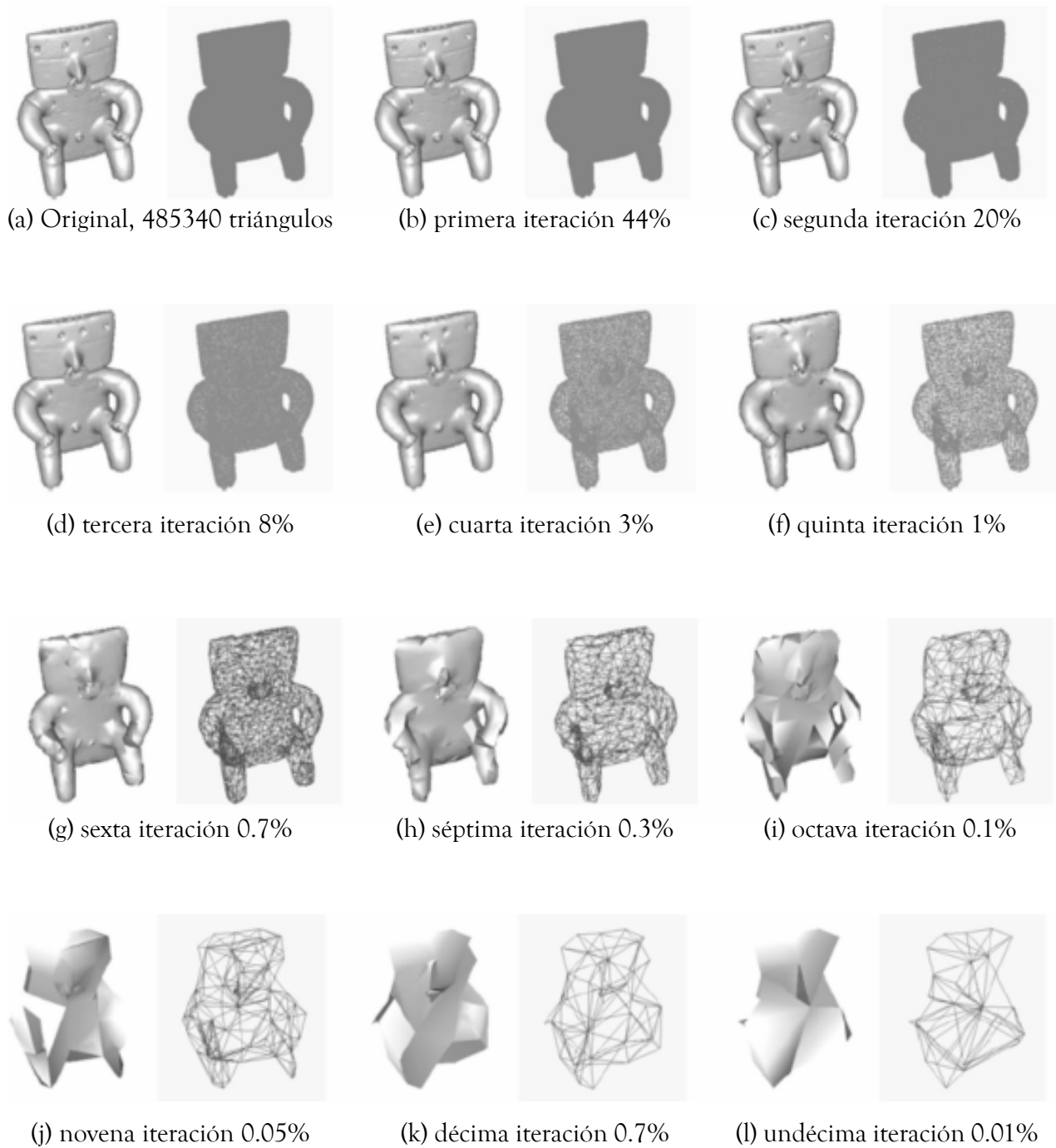


Figura 8: Colapso de triángulos realizado sobre el modelo de una escultura precolombina

RESULTADOS

Las pruebas realizadas fueron hechas en un PC con procesador Intel pentium III, de 789 MHz y memoria RAM de 128 MB. El sistema operativo utilizado fue Windows xp Pro. Los equipos desde los cuales se accede a la plataforma deben tener instalado un explorador de internet, un plugin para la visualización de elementos .wrl y la máquina virtual de Java.

La base de datos cuenta con el algoritmo de multirresolución descrito en la tercera sección del presente artículo, para la decimación de las imágenes de rango almacenadas; dicho algoritmo sólo se aplica para el proceso de visualización de los modelos en la Web, sin embargo, las imágenes completas se pueden descargar, teniendo en cuenta el aumento en el gasto computacional. En la figura 9, se pueden observar los 43 modelos que componen la base de datos.



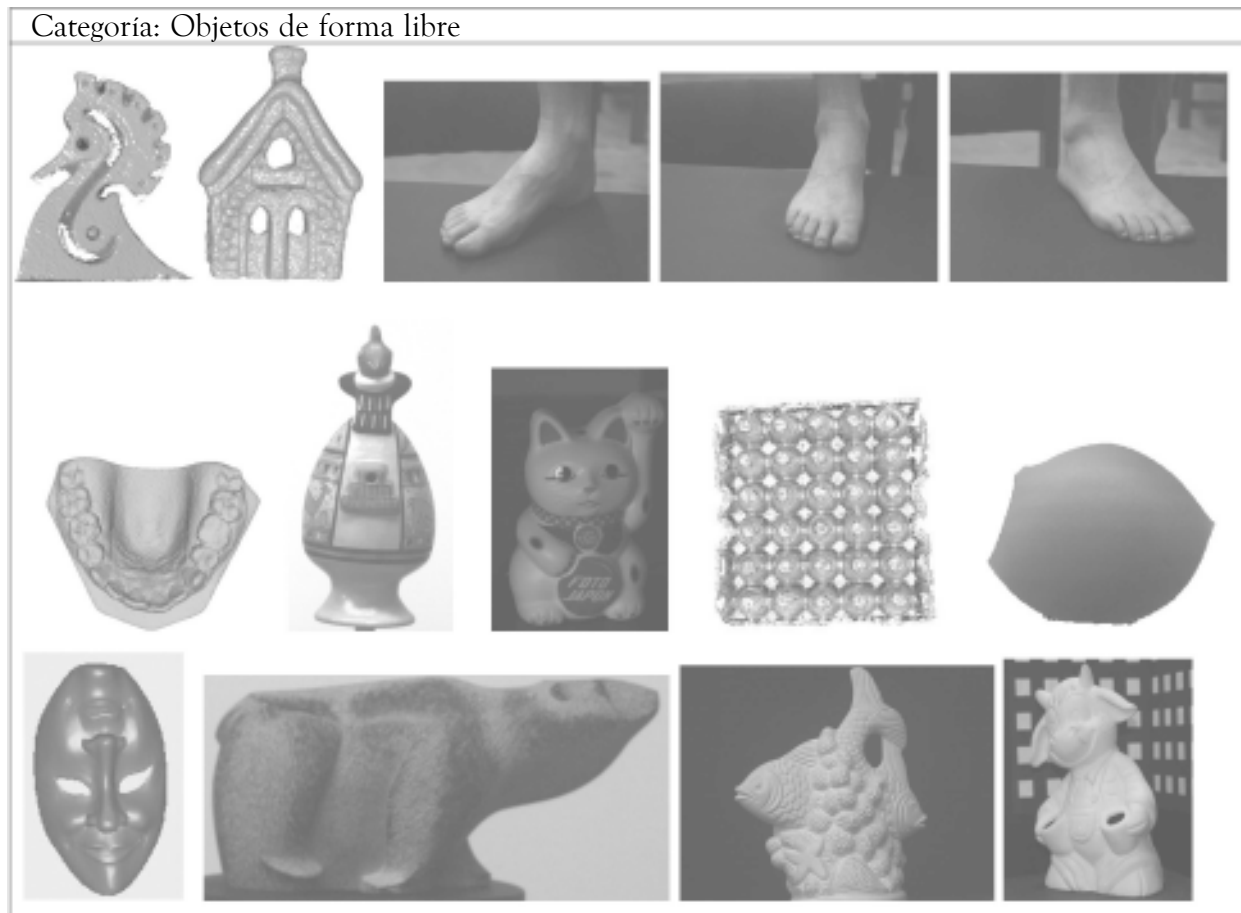


Figura 9: Objetos en la base de datos

Al aplicar el algoritmo de multirresolución, se decidió definir un número de 50000 triángulos como valor máximo, para todos los modelos. Dadas las pruebas mostradas en las figuras 7 y 8, y el promedio de número de triángulos de las imágenes de rango almacenadas en la base de datos, dicho valor suministra una buena resolución para la visualización en Web. En la ta-

bla 3, se observa el número de triángulos del modelo inicial y el número de triángulos del modelo decimado, así como el porcentaje de decimación. En algunos casos, el porcentaje de decimación es muy alto como para el modelo dental (68%). Mientras que en los modelos precolombinos y el gato de Foto-Japón, el porcentaje de decimación es más bajo.

Tabla 3: Resultados del algoritmo de multirresolución

Nombre del modelo	# triang. Iniciales	# triang. finales	% decimación
modelo1	444350	40183	9,043
modelo2	514444	47130	9,161
modelo3	485340	41251	8,499
san_agus	24767		
rostro_isabel	222463	40028	17,993
rostro_ernesto	37472		
rostro_flavio	21386		
rostro_jorge	207583	37050	17,848
rostro_arias	191028	48375	25,324
rostro_alex	152970	42403	27,720
rostro_alfredo	143430	37255	25,974
rostro_amanda	201203	37762	18,768
rostro_fabian	149734	40098	26,779
rostro_fabio	24326		
rostro_felipe	91007	43670	47,985
rostro_felija	197429	38568	19,535
rostro_jesus	162941	44343	27,214
rostro_juanb	209222	38696	18,495
rostro_luisa	22854		
rostro_oscar	137071	39546	28,851
rostro_osmarh	767463		
rostro_pierre	192866	41660	21,600
rostro_rafael	209365	44365	21,190
rostro	101400	39526	38,980
rostro_silvana	29294		
modelo4	189052	40339	21,338
modelo5	317505	39736	12,515
Vaca_toda	727384		
caballo_de_mar	40457		
Gato_fjapon	784780	43632	5,560
pescado	115042	45054	39,163
dentales	71941	49047	68,177
Casa	182755	37938	20,759
huevo	96150	36172	37,620
leonisa	98707	41672	42,218
máscara	233800	47102	20,146
Pie_toma0	140512	46775	33,289
Pie_toma1	139720	48321	34,584
Pie_toma2	124575	57859	46,445

En la medición de tiempos de visualización para los modelos completos, se encontró que la plataforma empleaba más tiempo en los modelos con mayor número de triángulos, como es el caso de los modelos precolombinos. Se presentaron algunos problemas con el algoritmo, en las imágenes de rango con número de triángulos mayor a 700000, ya que éstos utilizaban demasiada memoria del computador para el proceso de multirresolución. Los modelos cuyo número inicial de triángulos es menor a 50000, no se

decimaron. Se observa en la columna de tiempo inicial de carga, que éste no supera 3 segundos, y por tanto, no se necesita un proceso de decimación para visualización. Se observa en la tabla 4, el tiempo inicial y final de carga, además el porcentaje de ahorro en costo computacional, éste último con un valor máximo del 96% en el modelo 2, dado el porcentaje de decimación y el número de triángulos finales del precolombino, mientras para el modelo dental, el porcentaje de disminución es de sólo el 0.297%.

Tabla 4: Ahorro en tiempo computacional

Nombre del modelo	t. inicial de carga (seg)	t. final de carga (seg)	% ahorro en tiempo
modelo1	26,562	1,321	95,026
modelo2	28,657	1,126	96,071
modelo3	27,585	1,246	95,484
san_agus	1,118		
rostro_isabel	15,943	1,154	92,76
rostro_ernesto	1,854		
rostro_flavio	1,187		
rostro_jorge	12,398	1,001	91,926
rostro_arias	10,653	1,363	87,21
rostro_alex	9,653	1,197	87,605
rostro_alfredo	9,543	0,965	89,883
rostro_amanda	11,753	0,986	91,614
rostro_fabian	9,764	1,255	87,147
rostro_fabio	1,685		
rostro_felipe	4,998	1,265	74,688
rostro_felija	10,118	1,057	89,555
rostro_jesus	9,987	1,473	85,251
rostro_juanb	12,876	0,932	92,763
rostro_luisa	1,257		
rostro_oscar	8,294	1,187	85,688
rostro_osmarh	35,327		
rostro_pierre	10,092	1,745	82,707
rostro_rafael	11,944	3,853	67,743
rostro	6,872	1,237	82,007
rostro_silvana	1,83		
modelo4	9,962	1,421	85,734
modelo5	18,933	1,524	91,953
Vaca_toda	32,875		
caballo_de_mar	2,855		

Continuación **tabla 4:** Ahorro en tiempo computacional

Nombre del modelo	t. inicial de carga (seg)	t. final de carga (seg)	% ahorro en tiempo
Gato_fjapon	38,295	2,912	92,395
pescado	8,285	3,152	61,951
dentales	3,836	3,825	0,297
Casa	9,745	1,356	86,085
huevo	5,321	1,235	76,794
leonisa	5,587	1,824	67,354
mascara	13,396	2,654	80,188
pie_toma0	6,956	2,314	66,735
pie_toma1	6,953	3,955	43,125
pie_toma2	5,872	3,325	43,377

Finalmente, se observa en la figura 10 un ahorro significativo en tiempo de carga para todos los modelos, especialmente aquellos con un gran número de triángulos inicial, gracias al proceso de multirresolución. Los modelos no sólo redujeron

el número de triángulos sino también la carga computacional y el tamaño en memoria, algunos de ellos pesaban hasta 30 MB y este valor se redujo hasta 2 MB.

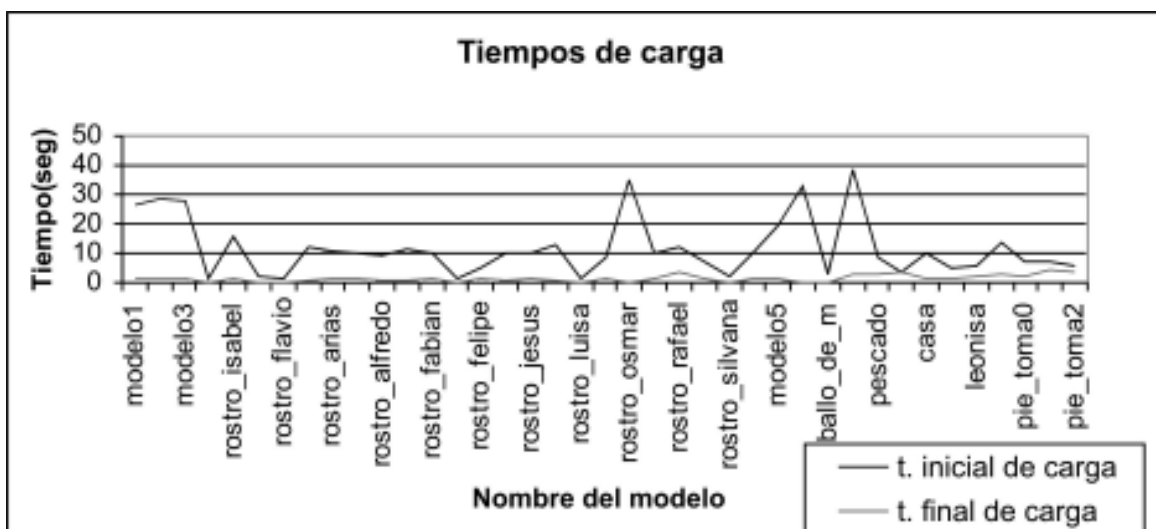


Figura 10: Comparación del tiempo de carga inicial con el tiempo de carga final

CONCLUSIONES

Se creó una plataforma Web, para la visualización, modificación y descarga de los modelos 3D que se encuentran en la base de datos; dicha BD está basada en un modelo relacional para la visualización se desarrolló un algoritmo de multiresolución basado en criterio de curvatura.

Se obtuvieron buenos resultados al calcular la desviación normal promedio para una imagen de rango y aplicar un algoritmo de decimación en aquellas zonas que presentan poca curvatura, para reducir el costo computacional de almacenamiento y de renderización conservando información de forma importante.

Aunque un algoritmo de decimación, sin aplicar un postproceso de triangulación, tiene falencias en cuanto a la forma de la malla, presenta ventajas en complejidad computacional y es adecuado para aplicaciones de renderización como plataformas web y juegos.

AGRADECIMIENTOS

Los autores agradecen a La Universidad Nacional de Colombia, quien a través de La Dirección Nacional de Investigación (DINAIN), financió parcialmente este trabajo en el marco del proyecto titulado *Modelado de superficies de forma libre empleando técnicas de visión artificial*.

BIBLIOGRAFÍA

- CANTZLER, H. 1997. An overview of range images.
- CASTRILLÓN, W. A. 2002. Segmentación de imágenes de rango a través de ajuste de superficies. Universidad Nacional de Colombia.
- CHACÓN, FD. 2000. Teoría de gráficos. Tesis de Maestría. Ciencias con Especialidad en Ingeniería en Sistemas Computacionales. Departamento de Ingeniería en Sistemas Computacionales, Escuela de Ingeniería. Universidad de Las Américas-Puebla
- CODD, E.F. 1970. A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM Vol 13 Nº 6 páginas 377–387*. Copyright © 1970, Association for Computing Machinery, Inc.
- GIENG, T. S. et al. 1997. Smooth hierarchical surface triangulations. Phoenix, Arizona.
- GÓMEZ, J.B. et al 2004. Reconstrucción de superficies a partir de imágenes de rango. Reporte técnico.
- KALVIN, A.D. et al. 1996. Surfaces: Polygonal mesh simplification with bounded error.
- KIM, T.H. et al. A review on level of detail. UTM Skudai, 81300 Johor, Malaysia.
- PAPAIOANNOU, G. et al. 2000. Segmentation and surface characterization of arbitrary 3d meshes for object reconstruction and recognition. In *Proceedings of the International Conference on Pattern Recognition*. University of Athens, GREECE, 2000.
- ROSSIGNAC, P. J. et al. 1993. Multi-resolution 3d approximations for rendering complex scenes.