

Multirresolución Adaptativa de Mallas Triangulares Basado en Criterios de Curvatura

Alexánder Ceballos* Jorge Hernández* Flavio Prieto*

Resumen

Los modelos 3D están compuestos por miles de polígonos y en ocasiones pueden obtenerse representaciones con la misma calidad visual pero con un menor número de polígonos, lo que hace que el costo computacional, de almacenamiento y de rendimiento disminuya. Este artículo propone un método para reducir el tamaño de imágenes 3D basadas en mallas triangulares, tratando de conservar la calidad visual al extraer la desviación normal promedio, una característica de forma, y simplificando la malla únicamente donde es baja.

Palabras Clave: *Multirresolución, curvatura, niveles de detalle, colapso de bordes.*

Abstract

The models 3D are compound for thousands of polygons and in occasions representations can be obtained with the same visual quality but with a smaller number of polygons, that makes that the cost computational, of storage and of rendering diminishes. This article proposes a method to reduce the size of 3D images based on triangular meshes, trying to conserve the visual quality extracting the average normal deviation, a form characteristic, and only simplifying the mesh where it is low.

Keywords: *Multiresolution, curve, detail levels, edge collapse.*

1 Introducción

La extracción de características de una superficie 3D se realiza principalmente para la segmentación y reconocimiento de objetos [11,2]. Entre las características principales se destacan la curvatura media y gaussiana, aunque también en los últimos años ha tenido auge el uso de wavelets, tanto para la simplificación de la superficie como para la caracterización [6,8,4].

* Departamento de Ingeniería Eléctrica, Electrónica y Computación, Universidad Nacional de Colombia Sede Manizales, {aceballosa, jehernandezl, faprietoo}@unal.edu.co

Uno de los mayores desafíos en visión artificial es conservar la calidad de las escenas reduciendo el costo computacional [5]. En imágenes 3D puede lograrse, por ejemplo, al reducir el número de polígonos, para aquellos objetos ubicados lejos del punto de vista del observador. También, para objetos adquiridos con diferente hardware, el sobremuestreo puede hacer que exista información redundante que puede ser simplificada sin afectar su calidad visual. Estas simplificaciones reciben el nombre de niveles de detalle [14].

Para la generación de los niveles de detalle, los algoritmos de multirresolución pueden ser de refinamiento, que comienzan con una aproximación tosca y van agregando detalles [10]; o de decimación, que empiezan con una superficie fina y van removiendo elementos [14]. También pueden ser clasificados como no poligonales (de modelos volumétricos [1], de modelos basados en imágenes [8], de superficies paramétricas, wavelets [12]) o poligonales (que son los más usados).

En la simplificación poligonal la idea es reducir el número de triángulos, vértices, bordes, huecos, túneles o cavidades, normalmente agrupando puntos y haciendo necesario casi siempre una nueva triangulación [5, 9, 13]. Existe una aproximación muy popular en la cual no es necesaria una nueva triangulación debido a que trabaja directamente sobre la malla poligonal llamada colapso de bordes. El colapso de bordes posee tres versiones, en las cuales la idea es remover puntos pertenecientes al mismo triángulo conservando la triangulación de los triángulos vecinos.

En el presente trabajo se pretende hacer uso de la desviación normal promedio, como característica para realizar simplificación de la malla en las regiones que consideramos brindan menos información (regiones planas), tratando de conservar aquellas más importantes en donde la forma local varía (las que presentan mayor curvatura).

El documento está organizado como sigue: la Sección 2 introduce el método usado para la simplificación de la malla y el criterio de curvatura usado para la conservación de información, la Sección 3 describe detalladamente los algoritmos usados en cada etapa, los resultados y su análisis son mostrados en las Secciones 4 y 5 respectivamente, y finalmente, las conclusiones son presentadas en la Sección 7.

2 Método Adaptativo de Multirresolución

El objetivo es reducir el número de puntos y triángulos de la malla, únicamente en aquellas regiones donde la curvatura es baja, para tratar

de conservar la calidad visual presente en la triangulación original. Por ejemplo, para rostros estas regiones son la nariz, la boca y los ojos, pero en general es aplicable a cualquier tipo de superficie que posea regiones más planas que brindan poca información comparadas con otras de alta variación.

Para lograr este objetivo es necesario obtener una medida de la curvatura local de la superficie, para lo cual se usan varias aproximaciones.

2.1 Estimación de la Curvatura

Después de implementar varias aproximaciones para la extracción de la curvatura como la curvatura gaussiana [2, 3], una medida similar a la redondez en imágenes 2D [10], la desviación del vector normal a un punto con los de los triángulos que lo comparten como vértice y la desviación normal promedio, se decidió emplear la desviación normal promedio que es una medida del ángulo entre el vector normal a un punto y el vector normal a la región a la que pertenece [11,3], (Figura 1), (Ecuación 3). Las expresiones de las curvaturas de algunas de las superficies geométricas consideradas como planos, cilindros, conos y esferas son relativamente sencillas. Sin embargo, las nubes de puntos son discretas y sería necesario obtener en primer lugar la superficie de aproximación, por lo tanto se emplearon versiones discretas. Se observó que los mejores resultados se obtuvieron con la desviación normal promedio, pues las demás aproximaciones dependían en gran medida de la densidad de la nube de puntos: por ejemplo, para la curvatura gaussiana y para nubes muy densas, es necesario calcular la derivada para observar diferencias entre regiones planas y curvas, mientras que para nubes no densas la diferencia entre regiones era clara antes de derivar. En el caso de la desviación normal promedio, se observaron resultados similares tanto para nubes densas como para las que no lo eran, y siendo el objetivo reducir la cantidad de puntos y triángulos en las regiones planas, lo que significa que la nube de puntos no tendría densidad uniforme, se decidió emplear esta aproximación como criterio de curvatura.

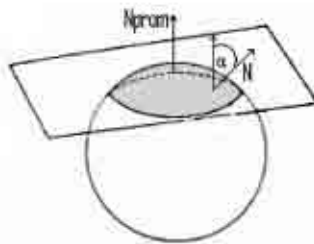


Figura 1: α es el ángulo entre el vector normal al punto y el vector normal a la región

Para superficies poliédricas que se fundamentan en nubes de puntos, es posible realizar el cálculo de curvaturas discretas locales, para lo cual se requiere la extracción de los vectores normales a la superficie en cada vértice del polígono [3]. Debido que las mallas triangulares definen planos para cada triángulo, es sencilla la obtención de la normal a cada uno y para la obtención de la normal a cada punto se promedian las normales de los triángulos adyacentes.

En un modelo 3D se puede definir una región superficial como una cara de la malla poligonal y consiste en una colección de polígonos conectados cuyas normales tienen orientación similar.

El vector normal promedio de una región puede ser visto como la suma de todos los vectores normales de los polígonos que forman la región ponderados por el área de cada uno. Debido a que se trata de un vector unitario, se debe dividir entre su magnitud como se muestra en la Ecuación 1, [11,3].

$$\vec{N}_{prom}(R_k) = \frac{\sum_{j|p_j \in R_k} A_j \vec{N}_j}{\left| \sum_{j|p_j \in R_k} A_j \vec{N}_j \right|} \quad (1)$$

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 (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:

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

Dada la definición de desviación normal promedio debe determinarse la región sobre la cual se realizará el análisis y debido que el interés no es segmentar las caras del modelo 3D, una buena aproximación puede ser hecha al considerar una región que rodee al punto. Una vecindad "sombrija" es una región compuesta por los triángulos que rodean a un punto (Figura 2). Para vecindades más grandes se puede expandir la definición dada al incluir aquellos triángulos que rodean la primera región.

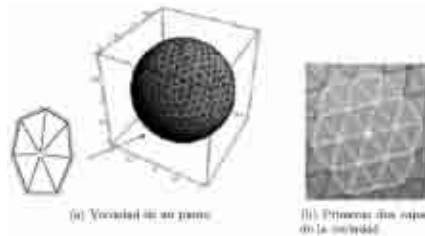


Figura 2: Región sombrilla

Es posible calcular la desviación normal promedio para cada punto de la nube si se posee información del vector normal a cada punto y además, se halla el vector normal a la vecindad de mismo.

2.2 Método de Multirresolución

El método implementado es el colapso de bordes debido a las ventajas que éste ofrece [7]:

La posición del nuevo vértice puede ser escogida con libertad, el nuevo vértice puede ser ubicado en el medio del borde, en un extremo e incluso en la mitad de un triángulo.

- La posibilidad de optimización escogiendo qué vértices son removidos dadas las características deseadas.
- No es necesaria una nueva triangulación, la malla varía sobre si misma al removerse los puntos con los bordes y triángulos ligados a ellos.
- Simplicidad, rapidez y efectividad, no se requiere gran cantidad de cálculos para escoger la nueva posición del punto, se trabaja directamente sobre los datos originales sin necesidad de transformaciones ni funciones complejas.

2.2.1 Colapso de Bordes

Dos vértices que forman un borde (puntos vecinos en el espacio) son unidos en uno solo, como se ilustra en la Figura 3. Al colapsar un borde se eliminan dos puntos pero se crea otro, además, se borran los dos triángulos adyacentes al borde y por lo tanto se reduce el número total de puntos en uno y el de triángulos en dos.

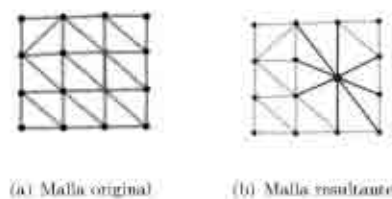


Figura 3: Colapso de bordes

Para colapsar bordes se debe tener en cuenta cuáles puntos pueden ser unidos. Considérese los siguientes casos:

- Colapsando todos los vecinos en un solo ciclo, y teniendo en cuenta que cada punto pertenece a más de un borde, no se simplifica la malla, lo único que se consigue es cambiar la posición espacial de los puntos, sin embargo, la complejidad de la malla es similar.
- Colapsando iterativamente todos los bordes, se obtiene un punto al final, debido a que cada punto nuevo pertenece a nuevos bordes, si no se mantiene información de cuáles ya han sido removidos se obtiene un resultado parecido al de la Figura 4.

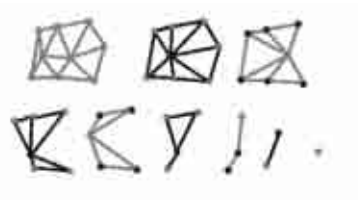


Figura 4: Colapso iterativo de bordes

Se vuelve necesario identificar los bordes nuevos para que no sean removidos. La manera más sencilla de conseguir esta información es identificar los puntos vecinos a los dos que fueron colapsados, pues son ellos junto con el nuevo punto los que conforman los nuevos bordes.

2.2.2 Semi Colapso de Bordes

Los vértices que conforman un borde son unidos, la posición del nuevo punto deberá ser la de uno de los vértices (Figura 5). Se debe tener el mismo cuidado que con el colapso de bordes para no terminar con un solo punto en el espacio. El número total de puntos y de triángulos es similar que en el colapso de bordes, aunque los cambios sobre la apariencia de la superficie son más notorios.

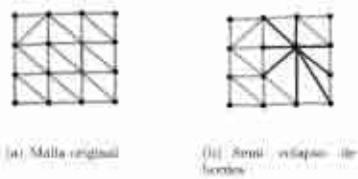


Figura 5: Semi colapso de bordes

2.2.3 Colapso de Triángulos

Los tres vértices de un triángulo son unidos en uno solo (Figura 6). Ésta es la única de las tres versiones cuya ubicación del nuevo punto no se encuentra sobre la superficie del modelo 3D, sin embargo, es la que mejor conserva la apariencia de la superficie. Igual que en el colapso y el semi colapso de bordes, el colapso de triángulos requiere cuidado

para no convertir la malla en un punto y para realmente decimarla. Al desaparecer el triángulo, los hacen también los tres triángulos adyacentes a sus lados, es decir que el número total de puntos decrece en dos mientras el de triángulos lo hace en cuatro (Figura 6).

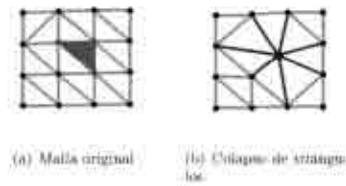


Figura 6. Colapso de triángulos

3 Implementación

La implementación se realiza como se muestra en el Algoritmo 1. Para la extracción de la curvatura se usa la Ecuación 3, contando con los vectores normales correspondientes a cada punto y hallando el vector promedio de la región que le rodea (Ecuación 2), para este caso se usan las tres capas de la vecindad (Figura 2). Una vez estimada la curvatura para cada punto de la nube, y con el fin de definir sobre cuáles se realizará el colapso, se usan dos estrategias, primero, se umbraliza el resultado en la media más dos veces la desviación estándar, y segundo, se hacen prescindibles el 50% de los puntos de la nube. Además, para la decimación se usan las tres aproximaciones descritas anteriormente y para la medida del error se propone el uso de la distancia de cada uno de los puntos del modelo original al triángulo más cercano de modelo decimado [14].

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: $\text{Bordes}(P, T)$

Paso 2:

Estimación de la desviación normal promedio: $\text{curvatura}(P, T)$

Paso 3:

Se simplifica la malla del modelo: $\text{decimación}(P, T)$

Paso 4:

Si se desea, volver al paso 1

A continuación se presenta cada uno de los procesos realizados en la implementación.

3.1 Extracción de Bordes

Se hace necesario la identificación de los puntos que conforman los bordes de la malla para evitar que sean removidos y se deforme (Figura 7). Se puede observar que para que un punto sea considerado borde su vecindad sombrilla debe estar incompleta. En una vecindad sombrilla completa, cada punto vecino al punto analizado debe estar compartido por mínimo dos triángulos. En el Algoritmo 2, para la obtención de los puntos vecinos al actual se incluyen en un arreglo los otros dos vértices de los triángulos que lo comparten, de forma tal que aquel punto que no se repita pertenece sólo a un triángulo de la vecindad sombrilla, lo que significa que es incompleta y por lo tanto el punto actual es un borde. Para la Figura 8, los puntos 1 y 3 sólo pertenecen a un triángulo en la vecindad sombrilla, además, se observa que la vecindad está incompleta y por lo tanto el punto actual es un borde.

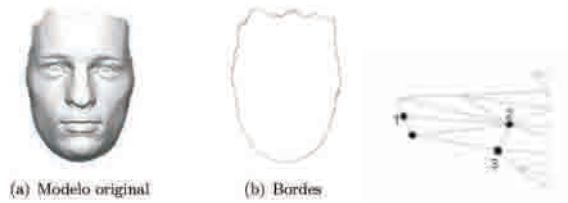


Figura 7: Bordés

Figura 8: Punto borde

Algoritmo 2 Bordés(P, T)

Entradas: conjunto de puntos $p_1, p_2, \dots, p_n \in P$.

Triangulación T de P .

Salida: puntos P conforman el borde.

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

Paso 1

Obtener los puntos vecinos a un punto p_i

Para todos los t_j **Hacer**

agregar a $vecinos_a$ b,c

agregar a $vecinos_b$ a,c

agregar a $vecinos_c$ a,b

Fin Para

Paso 2:

Si $vecinos_{p_i}$ es vacío **Entonces**

p_i es un borde

Fin Si

Paso 3:

Para todos los $p_k \in vecinos_i$

Si p_k no se repite en $vecinos_{p_i}$

p_i es un borde

Fin Si

Fin Para

3.2 Extracción de la Curvatura

En la implementación y al igual que para la extracción de bordes, el análisis de curvatura se lleva a cabo para cada punto de la malla triangular (Algoritmo 3). Se calcula la desviación normal promedio según la Ecuación 3. Para la obtención de la normal al punto se promedian las normales de los triángulos adyacentes. En la Figura 9, se observa que el análisis se implementa sobre tres capas de una región sombrilla, y para el cálculo del vector normal a ésta se usa la Ecuación 1. Finalmente, y para obtener únicamente dos valores discretos de curvatura (baja o alta), se umbraliza en el promedio más dos veces la desviación estándar, o se hacen prescindibles la mitad de los puntos, aquellos que tienen curvatura más baja, al organizar primero los valores de menor a mayor, y después, umbralizar la curvatura en el valor del medio.

Algoritmo 3 curvatura(**P**, **T**)

Entradas: conjunto de puntos $p_1, p_2, \dots, p_n \in P$.

Triangulación T de P .

Salida: curvatura C de cada punto.

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

Paso 1

Se calculan todas las normales de los triángulos t

$$N_i = (a_i - b_i) \times (a_i - c_i)$$

Paso 2:

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

Paso 3:

Se determina la región sombrilla de tercer orden de 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 4:

Se calcula el vector normal a la vecindad sombrilla de cada punto p_j .

$$\vec{N}_{Vecindad_j} = \left[\sum_k A_k \vec{N}_k \right]^{-1} \cdot \sum_k A_k \vec{N}_k$$

donde A_k y N_k son el área y el vector normal de cada triángulo perteneciente a la vecindad.

Paso 5:

Se calcula la desviación normal promedio para cada punto p_j como el ángulo entre su vector normal y el de su vecindad $Vecindad_j$.

$$\theta_j = \cos^{-1} \left(\vec{N}_j \vec{N}_{Vecindad_j} \right)$$

Y se umbraliza en el promedio más dos veces la desviación estándar ($c_j = 0$ ó $c_j = 1$)

O se organiza la desviación normal promedio de menor a mayor, y se umbraliza en el valor de la mitad ($c_j = 0$ ó $c_j = 1$).

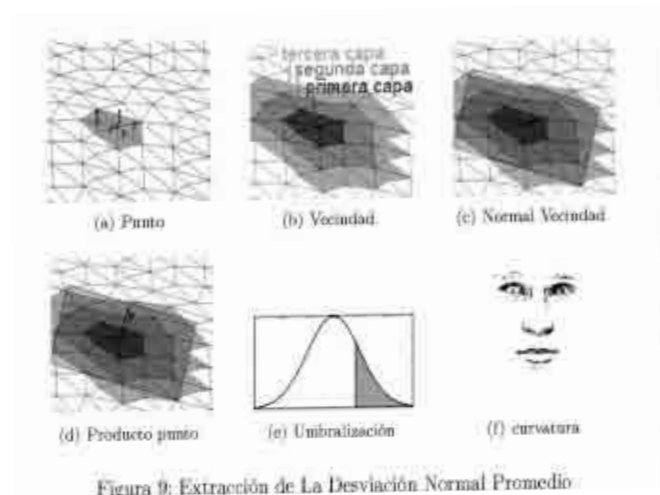


Figura 9: Extracción de La Desviación Normal Promedio

3.3 Decimación

La decimación consiste en la simplificación de la malla removiendo primitivas. Se remueven triángulos y puntos usando las tres versiones de colapso de bordes como se sugiere en el Algoritmo 4. Para mantener la forma de la malla y conseguir el objetivo propuesto, no se simplifican aquellos puntos que son bordes de la malla o cuya curvatura es alta. Debido a que el colapso debe ser hecho sobre puntos vecinos que pertenecen a un mismo triángulo, el análisis se hace para cada triángulo de la malla.

Algoritmo 4 decimación (P, T, C, B)

Entradas: conjunto de puntos $p_1, p_2, \dots, p_n \in P$.

Triangulación T de P .

Curvatura de cada punto C

Puntos bordes B

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

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

Paso 1

Para todos los triángulos t_i **Hacer**

Si ni a ni b ni c son bordes **y** a, b y c tienen curvatura baja **y** t_i puede ser removido **y** no ha sido borrado **Entonces**

$$d = (a + b + c) / 3$$

a, b y c son reemplazados por d

t_i ha sido borrado

todos los triángulos que compartían un punto con t_i no pueden ser removidos

Los triángulos que compartían dos puntos con t son borrados

Fin Si

Fin Para

Paso 2:

se corrige la malla del modelo decimado: corregir(P, T)

Para evitar simplificar la malla a un solo punto (Figura 4) se usan variables booleanas que definen cuáles ya han sido removidos, y lo más importante, cuáles no pueden ser borrados.

Al realizar un colapso de triángulos se eliminan los tres vértices que lo componen y los triángulos que poseen dos de ellos, se conservan los triángulos que sólo poseían uno de los puntos colapsados y no deben ser removidos en la siguiente iteración como se puede observar en la Figura 10.

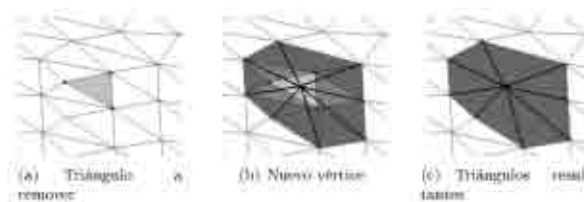


Figura 10: Simplificación de malla

Debido a que se modifica la malla sobre sí misma, se presentan algunos casos particulares en los que la malla se deforma de manera indeseable. Algunas veces, dos triángulos vecinos después de todo el proceso de decimación, terminan compartiendo los tres vértices, pareciendo una espiga que sobresale de la malla y por lo tanto ambos deben ser removidos. También, existe la posibilidad de que se forme una especie de pirámide, por ejemplo, en la Figura 11 se muestra la malla antes de ser simplificada (color rojo), y la malla decimada (color azul) en la cual se forma un triángulo que parece estar en orientación contraria a la de los que le rodean (color amarillo), esto puede ser solucionado al eliminar tres triángulos y reemplazarlos por uno sólo. Sin embargo, aún se hace necesario cambiar la orientación de algunos triángulos cuya orientación es contraria debido a la manipulación de sus vértices, se hace fácilmente al cambiar el orden en que lo componen los puntos (Algoritmo 5).

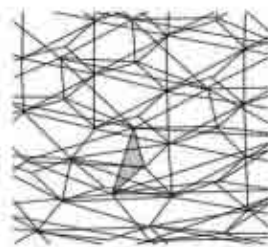


Figura 11: Corrección de malla

Algoritmo 5 corregir (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.Un triángulo t tiene 3 vértices, $t=(a,b,c)$ **Paso 1**

Remover espigas

Para todos los triángulos t_i **Hacer** **Si** algún triángulo vecino posee los mismos puntos **Entonces**

ambos son borrados

Fin Si**Fin Para****Paso 2:**

remover pirámides

Para todos los puntos p_i **Hacer** **Si** pertenece a tres triángulos y el ángulo entre uno de los triángulos y los otros dos es mayor a 90° **Entonces**

borrar los tres y conformar uno nuevo

Fin Si**Fin Para****Paso 3:**

Cambiar la orientación de los triángulos que están al revés

4 Resultados

Los experimentos se realizan primero sobre una imagen de rango de un rostro que posee 152970 triángulos, después se hace colapso de triángulos sobre modelos de 18904 y 23922 triángulos, todos adquiridos con el scanner vivid9i.

Se hacen dos aproximaciones para la umbralización de la desviación normal promedio. Primero, una estadística en la que se usa como umbral el valor promedio más dos veces la desviación estándar y con la cual se pretende conservar la información que consideramos más relevante, es decir la que posee curvatura más alta. Segundo, usando como umbral el valor obtenido al ordenar los valores y escoger la mediana, de esta forma se conserva más información, y más importante, se van simplificando las regiones más planas, en vez de conservar sólo las más curvas. En esta sección se muestran los resultados obtenidos para ambos métodos y en la Sección 4.4 se comparan el error obtenido y la convergencia.

4.1 Colapso de Bordes

4.1.1 Usando la desviación estándar

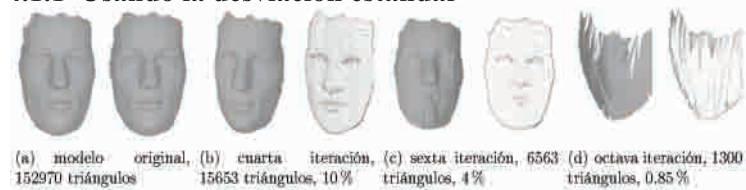


Figura 12: Colapso de bordes realizado sobre una imagen de rango de 152970 triángulos.

En la Figura 12 se pueden observar algunos resultados de la decimación iterativa sobre un modelo, se aprecia como la densidad de puntos es mayor en aquellas regiones donde la curvatura es mayor. La calidad visual sigue siendo aceptable en la cuarta iteración con el 10% de los triángulos del modelo original y en las demás iteraciones se degrada notablemente. La malla en las últimas iteraciones es irregular pues el ancho y alto de los triángulos no es proporcional, lo que hace que la calidad visual se reduzca drásticamente.

4.1.2 Umbralizando en el 50 %

Los resultados al poderse simplificar el 50% de los puntos (aquellos de menor curvatura), se muestran en la Figura 13. Al ser comparados con los obtenidos al umbralizar en el promedio más dos veces la desviación estándar, se observa que los primeros conservan más información en las regiones de alta variabilidad. Se aprecia que la calidad visual sigue siendo aceptable en la séptima iteración con el 31% de los triángulos del modelo original. En las demás iteraciones, al igual que en el caso anterior, la malla se degrada notablemente y se vuelve irregular, pues el ancho y alto de los triángulos no es proporcional, lo que hace que la calidad visual se reduzca drásticamente.

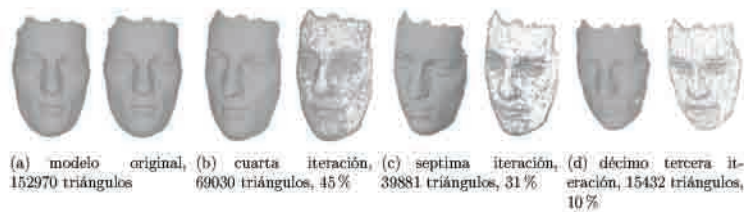


Figura 13: Colapso de bordes realizado sobre una imagen de rango de 152970 triángulos.

4.2 Semi-Colapso de Bordes

4.2.1 Usando la desviación estándar

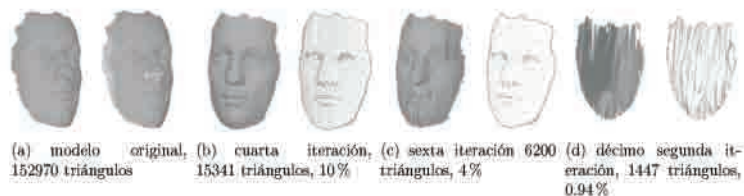


Figura 14: Semicolapso de triángulos realizado sobre una imagen de rango de 152970 triángulos.

Los resultados mostrados en la Figura 14 son similares a los obtenidos con el colapso de bordes; en la cuarta iteración la calidad visual del modelo aun es buena con el 10% de los triángulos del modelo original. La malla se degrada irregularmente al no mantenerse en relación el ancho y el alto de los triángulos. También se observa que se conservan más puntos en las regiones curvas.

4.2.2 Umbralizando en el 50 %

Los resultados mostrados en la Figura 15 muestran el modelo al ser decimado, usando como algoritmo de simplificación el semi-colapso de bordes, y umbralizando la curvatura para permitir la remoción del 50% de los puntos. Los resultados son similares a los obtenidos con el colapso de bordes; en la sexta iteración la calidad visual del modelo aun es buena con el 32% de los triángulos del modelo original. La malla se degrada irregularmente al no mantenerse en relación el ancho y el alto de los triángulos. Esta aproximación conserva más puntos de las zonas con alta curvatura que cuando se umbraliza en el promedio más dos veces la desviación estándar.

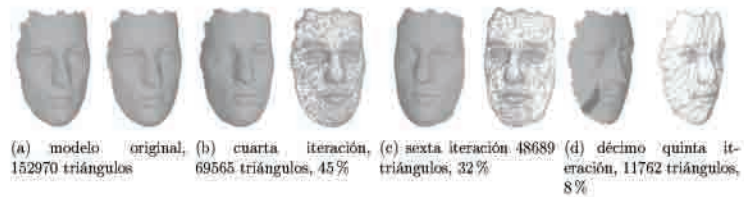


Figura 15: Semicolapso de triángulos realizado sobre una imagen de rango de 152970 triángulos.

4.3 Colapso de Triángulos

4.3.1 Usando la desviación estándar



Figura 16: Colapso de triángulos realizado sobre una imagen de rango de 152970 triángulos.

El colapso de triángulos es el método de multirresolución implementado que mejores resultados arroja como se puede ver en la Figura 16; en la cuarta iteración con el 6% de los triángulos la calidad visual sigue siendo relativamente buena. La malla se degrada de forma regular al mantenerse en proporción el ancho y alto de los triángulos, lo que hace que se logre una mayor grado de decimación con mejores resultados.



Figura 17: Colapso de triángulos realizado sobre una imagen de rango de 18904 triángulos.

Para este método de multirresolución se presentan también los resultados tras aplicar el algoritmo a un modelo de 18904 triángulos (Figura 17) y a un modelo de 23922 triángulos (Figura 18), en ambos se observa que los puntos se conservan más en las regiones de alta variabilidad que en aquellas planas.

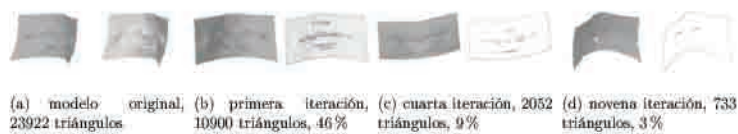


Figura 18: Colapso de triángulos realizado sobre una imagen de rango de 23922 triángulos.

4.3.2 Umbralizando en el 50 %

De nuevo el colapso de triángulos es el mejor método de multirresolución comparado con el colapso de bordes y el semi-colapso de bordes (Figura 19). En la sexta iteración con el 18% de los triángulos, la calidad visual sigue siendo relativamente buena. Comparado con la aproximación al umbralizar en el promedio más dos veces la desviación estándar, se ve que ésta conserva más información en las regiones curvas.

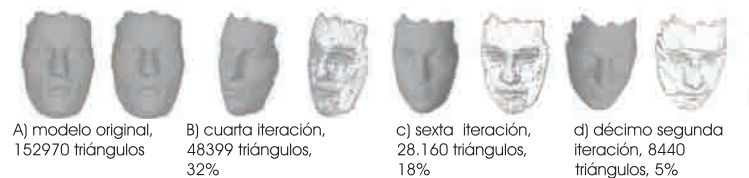


Figura 19: Colapso de triángulos realizado sobre una imagen de rango de 152970 triángulos.

Para la simplificación de la malla usando el colapso de triángulos se muestra, además, los resultados tras aplicarlo a un modelo de 18904 triángulos (Figura 20) y a uno de 23922 triángulos (Figura 21). Se observa que al hacer que el 50% de los puntos puedan ser removidos se conserva más información en las zonas curvas que al umbralizar en el promedio más dos veces la desviación estándar.



Figura 20: Colapso de triángulos realizado sobre una imagen de rango de 18904 triángulos.

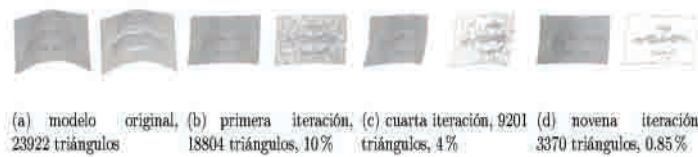


Figura 21: Colapso de triángulos realizado sobre una imagen de rango de 23922 triángulos.

4.4 Análisis de los Resultados

Para la medida del error se propone usar la distancia de los puntos del modelo decimado a los triángulos del original. Se usaron 3 modelos y se hicieron 30 iteraciones para cada una de las versiones de colapso de bordes y tanto para la umbralización de la curvatura en el promedio más dos veces la desviación estándar, como para cuando el 50% de los puntos pueden ser removidos. Para que las distancias sean comparables se normalizaron estadísticamente todos los modelos.

Los resultados se muestran en la Figura 22 y en la Figura 23. Como referencia se utilizaron los datos obtenidos con el Polygon Editor, un software especializado. Aunque el menor error fue obtenido con el Polygon Editor, se puede observar en los resultados que el colapso de bordes, el semi-colapso de bordes y el colapso de triángulos son buenas aproximaciones, además, el algoritmo propuesto conserva más información en las zonas curvas y puede definirse el umbral y así el nivel de detalle que se desea conservar.

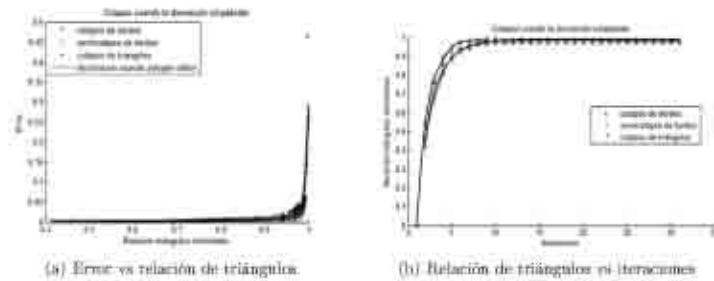


Figura 22: Usando la desviación estándar.

El algoritmo en todas sus versiones converge rápidamente cuando se umbraliza en el promedio más dos veces la desviación estándar, como se observa en la Figura 22, en la décima iteración ya se ha removido más del 90% de los triángulos originales para este caso, y en las iteraciones siguientes el número se reduce lentamente. Cuando se pueden eliminar el 50% de los puntos (aquellos de menor curvatura), el algoritmo converge más lento que en el caso anterior, debido a que conserva más información. También es notable que la mejor versión del colapso de bordes para ambas aproximaciones es el colapso de triángulos pues presenta el menor error y converge más rápido.

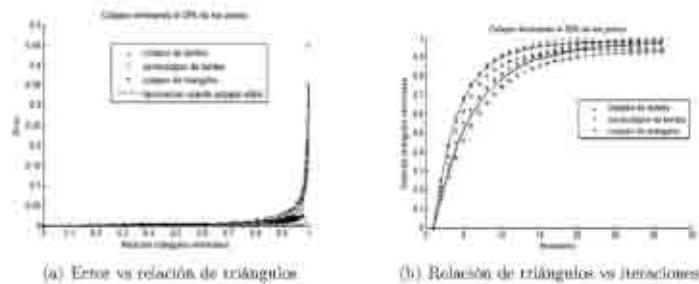


Figura 23: eliminando el 50% de los puntos

5 Conclusiones

Aplicando un algoritmo de decimación sobre imágenes de rango en las regiones más planas se puede reducir el costo de almacenamiento y de renderización conservando la calidad visual. Para definir qué regiones son planas o curvas una buena aproximación puede ser hecha con la desviación normal promedio como medida de forma local.

También puede definirse el nivel para la conservación de los detalles al definir el umbral para la desviación normal promedio, lo que hace que el algoritmo sea adaptable a los requerimientos de detalle de la aplicación.

Aunque un algoritmo de decimación en el cual no es necesario un postproceso de triangulación presenta 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.

6 Agradecimientos

Los autores agradecen a la Universidad Nacional de Colombia, quien financió parcialmente este trabajo a través del proyecto titulado “Modelado de Superficies de Forma Libre Empleando Técnicas de Visión Artificial”.

Referencias

- [1] S. Bayona and M. García. Estudio y Análisis de la Teoría de la Multirresolución en el Modelado de Sólidos. Tema : Técnicas de Representación y Modelado de Sólidos. Ingeniería Informática. Proceedings of the International Conference on Pattern Recognition, 2001-2002.
- [2] C. Bernal and M. Domínguez. Reconocimiento y Modelización automatizada de Instalaciones Industriales Normalizadas mediante Procesos de Ingeniería Inversa.
- [3] W. Castrillón. Segmentación de Imágenes de Rango a través de Ajuste de Superficies. Universidad Nacional de Colombia, 2002.
- [4] S. Castro, L. Castro, and A. DeGiusti. Multiresolution Volume Representation: The Wavelet Approach. Departamento de La Computacion, Universidad Nacional del Sur. Departamento de Matemáticas, Universidad Nacional del sur. Universidad Nacional de La Planta. Argentina. 2-4.
- [5] D. Chacón. Teoría de Gráficos. Tesis de Maestría. Ciencias con Especialidad en Ingeniería en Sistemas Computacionales. Proceedings of the International Conference on Pattern Recognition, Departamento de Ingeniería en Sistemas Computacionales, Escuela de Ingeniería. Universidad de Las Américas-Puebla, Mayo de 2000.
- [6] M. Garland and E. Shaffer. A Multiresolution Representation for Massive Meshes. IEEE Transactions on Visualization and Computer Graphics, Vol. 11, No. 2., Departamento de Ingeniería

en Sistemas Computacionales, Escuela de Ingeniería. Universidad de Las Américas-Puebla. 2, 2005.

- [7] T. Gieng, B. Hamann, K. Joy, G. Schlussmann, and I. Trotts. Smooth Hierarchical Surface Traingulations. Yagel, R. and Hagen, H. eds. IEEE Visualization. 379-386., Phoenix, Arizona 1997.
- [8] X. Gu, S. Gortler, and H. Hoppe. Geometry Images. Harvard University and Microsoft Research. 2.
- [9] D. Kalvin and R. Taylor. Superfaces: Polygonal Mesh Simplification with Boinded Error. IEEE Computer Graphics and Applications. 64-77., Departamento de Ingenieria en Sistemas Computacionales, Escuela de Ingeniería. Universidad de Las Américas- Puebla., 1996.
- [10] D. Mery. Visión por Computador. Universidad Católica de Chile., 17 de agosto de 2004.
- [11] G. Papaioannou, E.-A. Karabassi, and T. Theoharis. Segmentation and Surface Characterization of Arbitrary 3D Meshes for Object Reconstruction and Recognition. Proceedings of the International Conference on Pattern Recognition, University of Athens, GREECE. 1-3, 2000.
- [12] L. Pastor and A. Rodríguez. Surface Approximation of 3D Objects from Irregularly Sampled Clouds of 3D Points Using Spherical Wavelets. Departamento de Ciencias Experimentales y Tecnológicas Universidad Rey Juan Carlos, Madrid, España. Departamento de Tecnología Fotónica, Universidad Politécnica de Madrid, Madrid, España. Proceedings of the International Conference on Pattern Recognition, Mayo de 2000.
- [13] J. Rossignac and P. Borrel. Multi-resolution 3d Approximations for Rendering Complex Scenes. Springer-Verlag. 455-465., Falcendo, B. and Kunii, T. L. eds. Modeling in Computer Graphics., 1993.
- [14] K. Tan and S. Daut. A Review on Level of Detail. Department of Graphics and Multimedia, Universiti Teknologi Malaysia. 3-5, UTM Skudai, 81300 Johor, Malaysia.