

RAY CASTING PARA LA DEFINICIÓN DE ZONAS DE INTERÉS EN SIMPLIFICACIÓN TOPOGRÁFICA

María V. Cifuentes^{a,b}, Juan P. D'Amato^{a,c}, Cristian García Bauza^a, Marcelo J. Vénere^a,
Pablo Lotito^{a,c}, Alejandro Clause^{a,c}

^aPLADEMA, CNEA y Universidad Nacional del Centro, 7000 Tandil, Argentina,
([@exa.unicen.edu.ar](mailto:cifuentes,jpdamato,rgarcia,venerem,plotito,clause))

^b Comisión de Investigaciones Científicas de la Provincia de Buenos Aires, Profesional de Apoyo

^c Consejo Nacional de Investigaciones Científicas

Palabras Clave: Modelos Topográficos, Ray casting, GIS, Procesamiento de imágenes.

Resumen. Se propone un algoritmo que reduce la cantidad de polígonos utilizados para representar superficies extensas, como la de modelos topográficos, de acuerdo a posibles trayectorias que siga un observador sin perder calidad en sus inmediaciones. Como indicador de error se emplea la curvatura local y el campo visual del observador, descrito por una máscara binaria generada automáticamente por un algoritmo de *ray casting* cuyos elementos actúan como factores multiplicativos de la curvatura local original del modelo. Además, se incluye una aplicación a modelos de elevación artificiales/reales.

1 INTRODUCCION

El tratamiento interactivo de grandes superficies topográficas reales tiene importantes aplicaciones en geografía, defensa, y juegos computacionales, entre otros. En algunas aplicaciones de realidad virtual topográfica sucede que ciertas regiones del escenario siempre permanecen alejadas u ocultas al observador. Un caso típico lo constituyen los escenarios navales (Cifuentes, D'Amato, García Bauza, Vénere , 2005), donde el espectador se encuentra en el interior de embarcaciones que transitan un lago rodeado por una topografía irregular y los posibles recorridos se restringen a ciertas regiones del mapa. En casos como este es conveniente evitar la sobrerrepresentación del terreno para acelerar la visualización. Dentro de esta área, en los últimos años se han desarrollado en Argentina varios algoritmos de simplificación automática de modelos digitales de elevación (MDE) basados en descomposiciones *quadtree* guiadas por indicadores de curvatura de la superficie (Cifuentes, Vénere y Clause , 2004) y posición del observador (Cifuentes, D'Amato, García Bauza, Vénere y Clause , 2005).

En el presente trabajo se presenta un criterio novedoso para la reducción poligonal de los modelos de elevación, introduciendo el concepto de campo de visión de un observador inmerso en la escena. Este campo consiste en una máscara binaria generada con técnicas de Ray Casting (Appel, 1968).

El trabajo se organiza de la siguiente forma. Primero se introduce al concepto de simplificación poligonal guiada por indicadores de error, a continuación, se detalla la metodología para construir el campo de visión y finalmente se muestran los resultados obtenidos al aplicar la técnica a una malla sintética.

2 SIMPLIFICACIÓN POLIGONAL

2.1 Metodología para la simplificación poligonal

La metodología de simplificación poligonal de superficies extensas se basa en la transformación del campo de alturas original en un campo cuadrado de longitud de lado $L/2^k$, donde k identifica el nivel de refinamiento y L el tamaño del dominio (Samet 1984). El algoritmo subdivide al nodo raíz (*breadth first*) hasta que se cumpla con un cierto criterio de tolerancia. Posteriormente se construye una jerarquía de mallas con complejidades diferentes organizadas en un *quadtree* (árbol cuaternario). Finalmente la multimalla resultante debe conformarse adecuadamente para eliminar huecos.

La última operación puede realizarse de varias maneras. En el presente caso se genera un *quadtree* restringido a cuadrantes adyacentes difiriendo en no más de un nivel jerárquico, tal como define (Ibaroudene, Demjanenko, y Acharya 1990). Los huecos remanentes en las uniones de este último tipo se resuelven usando un diccionario de soluciones estándar (Von Herzen y Barr 1987) La triangulación que se obtiene en esta manera presenta sólo cuatro tipos de triángulos, de excelente calidad.

2.2 Indicador de curvatura

El primer indicador guía de la simplificación de mallas es la curvatura del terreno. La curvatura local compara la desviación de la geometría intrínseca de la superficie respecto a la geometría del plano. La idea del criterio es que una región con alta curvatura debe ser modelada con más densidad de triángulos que una región relativamente más plana. De esta forma los lugares con baja curvatura pueden ser simplificados sin perder calidad visual.

La técnica de simplificación consiste en dividir en cuatro recurrentemente toda región cuya

curvatura acumulada supere un umbral de tolerancia permitido por el usuario (ec. 1), es decir una región será dividida en cuatro subregiones si cumple:

$$\kappa_R = \sum_{i=1}^n \kappa_i \leq T \quad (1)$$

donde κ_R es la suma de las curvaturas locales de la región.

La curvatura local κ se calcula numéricamente mediante un esquema centrado en diferencias finitas sobre el campo de alturas $h_{i,j}$. De acuerdo a este último, el gradiente se calcula como:

$$G_x^{i,j} = \frac{\partial h_{i,j}}{\partial x} = \frac{h_{i+1,j} - h_{i-1,j}}{2\Delta x} \quad \text{y} \quad G_y^{i,j} = \frac{\partial h_{i,j}}{\partial y} = \frac{h_{i,j+1} - h_{i,j-1}}{2\Delta y} \quad (2)$$

donde Δx y Δy representan la distancia entre los pares de puntos vecinos a las coordenadas (i,j) .

De la misma manera, el tensor de curvatura (o matriz Hessiana) κ_{ij} resulta:

$$\kappa_{ij} = \begin{bmatrix} (G_x^{i+1,j} - G_x^{i-1,j}) & (G_y^{i+1,j} - G_y^{i-1,j}) \\ (G_x^{i,j+1} - G_x^{i,j-1}) & (G_y^{i,j+1} - G_y^{i,j-1}) \end{bmatrix} \quad (3)$$

Dado que para los efectos de visualización interesa conocer sólo una curvatura promedio, se utilizó como indicador de curvatura local la suma de los cuadrados de los elementos de κ_{ij} (Cifuentes, D'Amato, García Bauza, Vénere y Clause , 2005), esto es:

$$\kappa_{ij}^2 = (G_x^{i+1,j} - G_x^{i-1,j})^2 + (G_y^{i+1,j} - G_y^{i-1,j})^2 + (G_x^{i,j+1} - G_x^{i,j-1})^2 + (G_y^{i,j+1} - G_y^{i,j-1})^2 \quad (4)$$

El indicador de curvatura definido en (ec. 4) puede pensarse como una norma del tensor de curvatura que mide la distancia de dicho tensor al cero, valor que indica planaridad. Este indicador es consistente en el sentido que está relacionado con la curvatura media y la curvatura gaussiana, de modo que si dicho indicador es cero ambas curvaturas también lo son. La curvatura media se define como el promedio de los autovalores de κ_{ij} y la curvatura gaussiana como su producto (ver Apéndice).

2.3 Simplificación guiada por máscaras

En algunas aplicaciones sucede que ciertos sectores de la superficie permanecen invisibles al observador por lo que resulta redundante su representación detallada. Un ejemplo típico se da en los escenarios navales, donde el observador ubicado en el interior de una embarcación visualiza la costa, que por la lejanía y la perspectiva forzosamente sólo se ve parcialmente. En este contexto es irrelevante el detalle excesivo en aquellas regiones que permanecen ocultas.

En estos casos, el criterio de simplificación de la malla soporte de la representación deberá ser preservar el detalle en las regiones visibles y destinar unos pocos polígonos para la representación de las regiones que permanezcan invisibles. Para lograr este objetivo se propone la construcción automática de una máscara determinada por un indicador de

visibilidad de cada punto de la malla. La implementación de la máscara en el proceso de simplificación se realiza introduciendo un factor de corrección en la [ec. 1](#), esto es:

$$v_n \kappa_R \leq T \quad (5)$$

donde v_n es un número real que establece la visibilidad del punto espacial desde la perspectiva del observador ($0 \leq v_n \leq 1$).

La máscara de visibilidad puede generarse automáticamente mediante un algoritmo de *ray casting*. En esta técnica se calculan las intersecciones del terreno con rayos salientes de la posición del observador. Para cada rayo, el punto de intersección más cercano al observador será visible ($v_n > 0$), mientras que las demás intersecciones se asignan con $v_n = 0$. En la [Figura 1](#) se esquematiza este comportamiento.

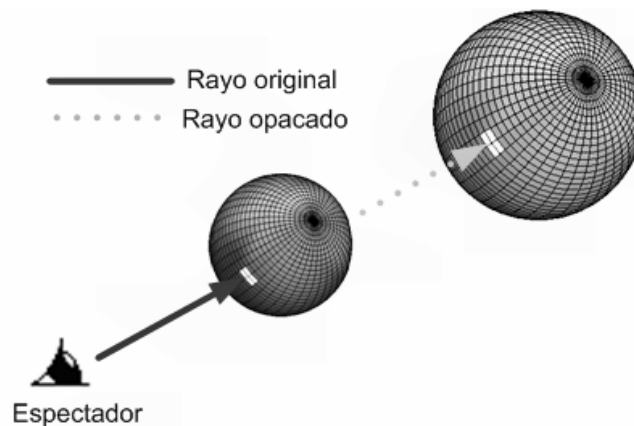


Figura 1: Esquema del algoritmo de RayCasting

Existen métodos optimizados para el cálculo de las intersecciones de los rayos con los elementos del MDE ([Challinger 1993](#)) (triángulos en nuestro caso), que evitan el incremento masivo de los cálculos mediante el ordenamiento inteligente de los elementos. No obstante, aplicar la técnica de *ray casting* no es trivial y a menudo es conveniente que la máscara binaria se construya antes del inicio de la aplicación considerando todas las posibles posiciones del observador. La máscara de visibilidad resultará entonces de aplicar el algoritmo en aquellos puntos que conforman el borde de la zona de movimiento del observador; los puntos interiores no aportan información adicional al tratarse del borde de una superficie y donde el movimiento es plano y de altura constante.

A continuación se listan los pasos principales para generar la máscara de visibilidad.

- definir la superficie de exploración
- en cada punto del contorno de la superficie de exploración
 - emitir rayos en todas las direcciones
 - obtener la lista de triángulos intersecados
 - ordenarlos por la distancia, para eliminar los triángulos ocultos
 - asignar el indicador de visibilidad a cada triángulo

En el algoritmo, el entorno visible de un espectador se determina mediante rayos dirigidos en todas direcciones con una separación angular de $\Delta\alpha$ para el acimut y $\Delta\beta$ para la elevación. Es importante determinar correctamente la tasa de cambio de la orientación del rayo; si se

elige un $\Delta\alpha$ muy grande, polígonos muy pequeños o muy alejados nunca serán intersectados por el rayo, tal como sucede en la [Figura 2](#).

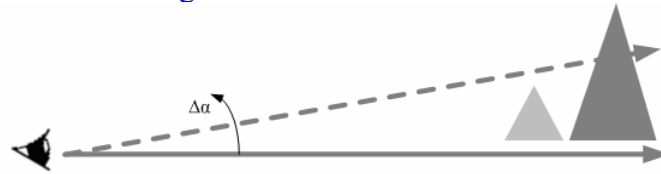


Figura 2: Caso de error en la definición del $\Delta\alpha$ de dos rayos consecutivos

Sin embargo tampoco se puede elegir dichos ángulos muy pequeños porque equivaldría a mayor tiempo de cálculo al tener que considerar más rayos. Los valores para $\Delta\alpha$ y $\Delta\beta$ máximos que garantizan no perder polígonos se puede calcular considerando todos los triángulos en perspectiva desde el observador. Cada triángulo estará encerrado por un rectángulo de base Δx y altura Δy , y estará separado del observador por una distancia r . Entonces,

$$\tan \Delta\alpha = \frac{1}{2} \min\left(\frac{\Delta x}{r} \geq \varepsilon\right), \quad \tan \Delta\beta = \frac{1}{2} \min\left(\frac{\Delta y}{r} \geq \varepsilon\right), \quad (6)$$

donde el mínimo se calcula sobre todos los cocientes que sean mayores o iguales que ε , para evitar triángulos que en perspectiva son degenerados. Los triángulos cuyo cociente en ambos casos es menor que ε serán probablemente descartados, por lo tanto su valor debe ser suficientemente chico.

3 RESULTADOS NUMÉRICOS

La [Figura 3](#) compara una simplificación en base a la [ec. 1](#) (sólo criterio de curvatura) de un MDE de 640.000 triángulos con la representación original. La malla simplificada está compuesta por 162.391 triángulos, y puede verse que se asignan elementos de mayor tamaño para representar las zonas más planas, mientras que las zonas de montañas tienen mayor detalle.

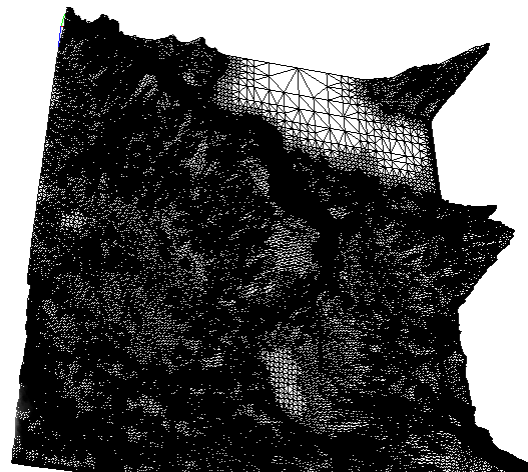
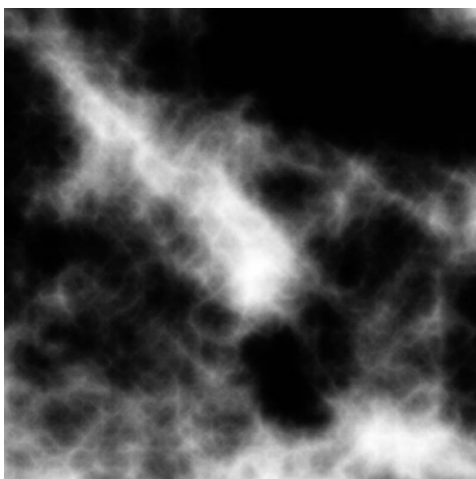


Figura 3: Simplificación basada en la curvatura local.

La [Figura 4](#) muestra una simplificación del mismo escenario introduciendo la corrección por visibilidad. A la izquierda, se muestra la máscara de visibilidad que segmenta al MDE en dos sectores: los polígonos invisibles al observador (en color negro) y los visibles (en color blanco). Esta máscara surge de aplicar el algoritmo de *ray casting* a los puntos del contorno de la zona marcada en rojo. A la derecha, se muestra la triangulación de 49.911 polígonos resultante. Nótese que, las regiones imperceptibles al ojo del observador como las ubicadas detrás de las grandes montañas usan menos polígonos de los necesarios para su representación. Estos resultados se deben a que el factor de visibilidad empleado redujo la curvatura local en todas aquellas regiones ubicadas fuera del campo de visión, y por lo tanto no se forzaron sub-divisiones desde algoritmo de simplificación. Además, variando el peso de los coeficientes de visibilidad con respecto a la curvatura local también puede variarse la densificación de la malla.

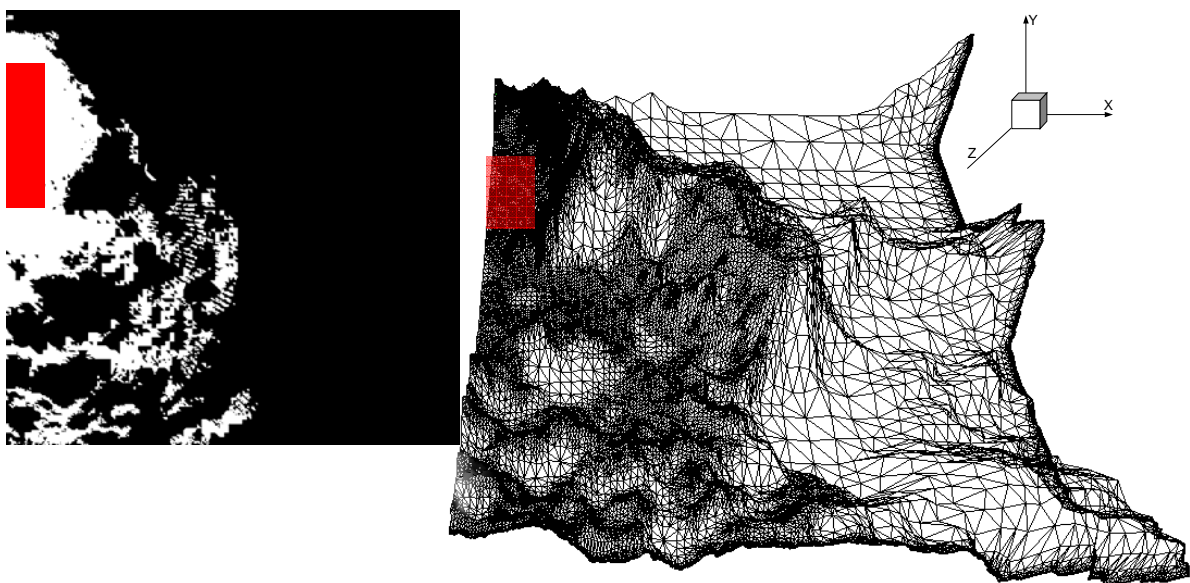


Figura 4: Simplificación guiada por máscaras de visibilidad.

Finalmente, se analizaron los resultados luego de incluirlos en un Escenario 3D ([Cifuentes, D'Amato, García Bauza, Vénere , 2005](#)). En la [Figura 6](#) se observa desde el interior de la superficie de exploración, dos mallas obtenidas por los dos criterios. Es posible observar como la aplicación de la máscara no varió la silueta de la malla.





Figura 6: Modelo aproximado (arriba) y original (abajo).

Comparativamente, en la [Figura 7](#) se observa el mismo MDE pero desde una posición fuera del horizonte del observador. Es claro notar la pérdida de definición de la malla para zonas con mucho cambio de pendiente.

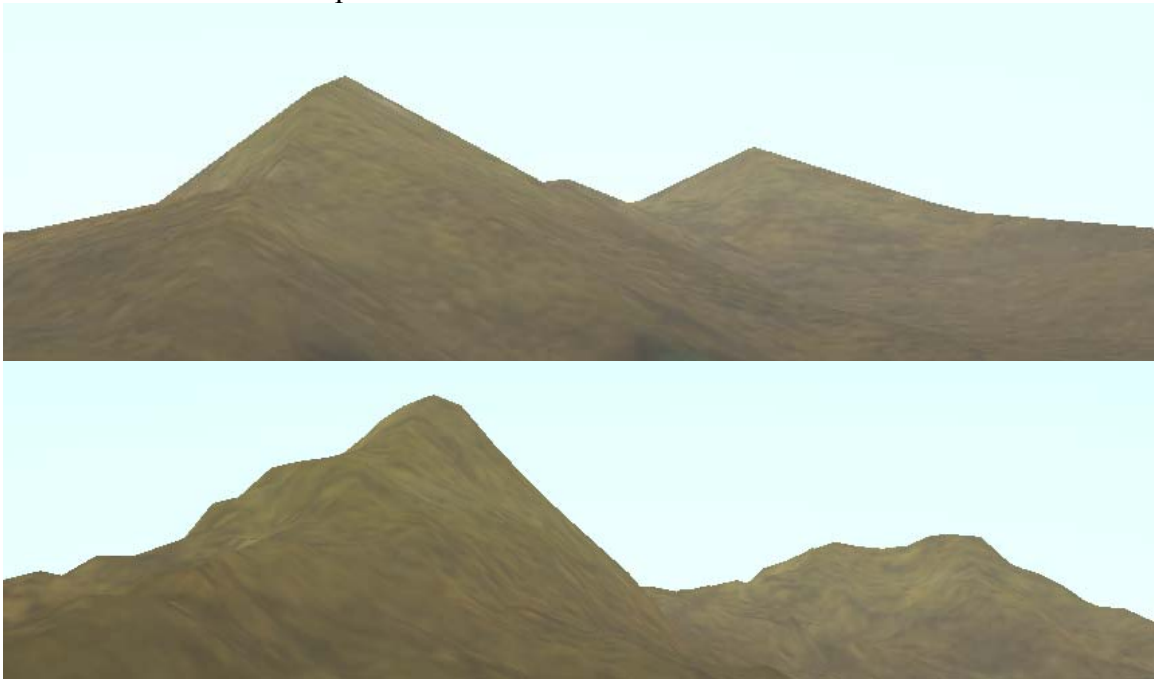


Figura 7: Modelo aproximado (arriba) y original (abajo).

El mismo concepto fue llevado exitosamente a un escenario real, correspondiente al lago Nahuel Huapi. El Nahuel Huapi es un espejo de agua rodeado por grandes montañas, usualmente navegado por pequeñas embarcaciones con la particularidad de que cualquier observador embarcado no puede ver más allá del área que rodea al lago.

4 CONCLUSIONES

Se presentó una técnica de simplificación de modelos digitales topográficos que aplica criterios de curvatura del terreno y visibilidad respecto del observador. La técnica propuesta en este trabajo significa un adelanto considerable con respecto a la velocidad de visualización manteniendo una representación aceptable del escenario.

La implementación de un algoritmo de Ray Casting con satisfactorios resultados, abre un interesante panorama para realizar la implementación de dicho algoritmo de forma más eficiente. La estructura de máscaras de visibilidad ha demostrado ser óptima para combinar con otros criterios de simplificación.

5 APÉNDICE

5.1 Relaciones entre los indicadores de curvatura

Si escribimos el tensor de curvatura como:

$$\kappa = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad (7)$$

entonces el indicador de curvatura usado para guiar la simplificación de mallas vale:

$$\kappa^2 = a^2 + b^2 + c^2 + d^2 \quad (8)$$

A su vez, el polinomio característico de la matriz está dado por:

$$(x - \lambda_1)(x - \lambda_2) = x^2 - (a + d)x + ad - bc \quad (9)$$

por lo que la curvatura media H y la curvatura gaussiana K valen:

$$\begin{aligned} H &= \frac{\lambda_1 + \lambda_2}{2} = \frac{a + d}{2} \\ K &= \lambda_1 \lambda_2 = ad - bc \end{aligned} \quad (10)$$

además es sabido que $K \leq H^2$, con lo que:

$$K \leq H^2 = \left(\frac{a + d}{2} \right)^2 \leq \frac{a^2 + b^2}{2} \leq \frac{a^2 + b^2 + c^2 + d^2}{2} = \frac{\kappa}{2} \quad (11)$$

donde la penúltima desigualdad proviene de la desigualdad entre la media aritmética y la media cuadrática.

La relación de desigualdades (8) muestra que si nuestro indicador es pequeño, también lo serán las curvaturas.

REFERENCIAS

Appel A.. On calculating the illusion of reality. *In IFIP congreso (2)*, pages 945-950, 1968.

Challinger J., Scalable Parallel Volume Raycasting for Nonrectilinear Computational Grids, *IEEE* pp 87-111, 1993.

Cifuentes, M.V., Dr. Vénere, M.J., Dr. Clause, A. Un algoritmo para la simplificación poligonal de modelos topográficos digitales. *33ª JAIHO, Jornadas Argentinas de Informática e Investigación Operativa. Córdoba, 2004.*

Cifuentes, M.V., Ing. D' Amato, J.P., Ing. García Bauza, C., Dr. Vénere, M.J., Dr. Clause,

- A. Remallado interactivo como método para la navegación eficiente de modelos topográficos.
MECOM 2005. Buenos Aires, Noviembre de 2005.
- Cifuentes, M.V., D' Amato, J.P., García Bauza, C., Vénere, M.J. Editor de escenarios para aplicaciones de Realidad Virtual. *34ª JAIIO, Jornadas Argentinas de Informática e Investigación Operativa*. Rosario, Septiembre de 2005.
- Ibaroudene, D, Demjanenko, V & Acharya, R. Adjacency algorithms for linear octree nodes. *Image and Vision Computing, Butterworth & Co Publishers Ltd.*, 8(2):115-123, May 1990.
- Samet, H. The quadtree and related hierarchical data structures. *Association for Computing Machinery. Computing Surveys*, 16 (2):187-260, June 1984.
- Von Herzen, B. & Barr, A. Accurate triangulation of deformed, intersecting surfaces. *Computer Graphics , SIGGRAPH'87 Proceedings*, pp. 103-110, 1987.