

## VISUALIZACIÓN INTERACTIVA ANISOTRÓPICA DE MODELOS TOPOGRÁFICOS

M. V. Cifuentes<sup>a,b</sup>, J. Dottori<sup>b</sup>, L. Lo Vercio<sup>b</sup> y A. Clause<sup>b,c</sup>

<sup>a</sup>Comisión de Investigaciones Científicas de la Provincia de Buenos Aires, Argentina

<sup>b</sup>PLADEMA-ISISTAN, Universidad Nacional del Centro, 7000 Tandil, Argentina  
({cifuentes,clause}@exa.unicen.edu.ar, {javiermdq,lucaslovercio}@gmail.com)

<sup>c</sup>Comisión de Energía Atómica, Argentina

**Keywords:** Visualización, Modelo digital de elevación, Procesamiento de imágenes, Mallas.

**Abstract.** Se proponen algoritmos *bottom-up* para la simplificación automática de modelos topográficos digitales en  $O(n)$ . El método general consiste en construir una *triangulación base*, que sirve de referencia a la visualización dinámica. Definido un indicador de curvatura local se zonifica la superficie en porciones rectangulares que acumulan una cota máxima de curvatura. Esta zonificación se organiza en un *quadtrees restringido* cuyos nodos terminales se triangulan en una de cinco posibles situaciones que son identificadas rápidamente gracias a la localización directa de regiones vecinas. El detalle conveniente en cada zona lo establece un nuevo indicador que integra la posición e inclinación de la vista del observador. De esta manera en cada instante se identifican las regiones que deben ser remalladas y luego se modifica la *triangulación base* agregando precisión al escenario en estas regiones. Los resultados obtenidos muestran una mejora de la velocidad de visualización en un sobrevuelo pasante sobre la superficie, comparando contra una estrategia de remallado dinámico completo de la escena.

### 1 INTRODUCCION

Los Modelos Digitales de Elevación (*MDE*) son representaciones de porciones de la superficie terrestre descrita por una grilla regular, que asigna una altura escalar a cada celda del terreno representando la elevación local respecto a un nivel de referencia. La densidad de la grilla determina la calidad de la representación y es usual encontrar modelos con varios millones de píxeles, por lo que la visualización flexible constituye un problema computacional de cuidado. Las visualizaciones tienen gran importancia en el desarrollo de simulaciones inmersivas de terrenos geográficos (Vénere *et al* 2005, Cifuentes *et al* 2006), como por ejemplo el sobrevuelo en un helicóptero virtual. Entre las técnicas de visualización de *MDEs* más conocidas se destacan la representación por niveles de detalles conformes al punto de vista del observador, la eliminación de regiones del terreno temporariamente invisibles, el empleo de técnicas que efectúan transiciones suaves en la geometría de la malla, etc. (Abásolo *et al* 2000, Balmelli *et al* 1999, Hoppe 1997, Lindstrom *et al* 2001, Gross *et al* 1996, Rottger *et al* 1998 y Pajarola 2002).

En Cifuentes *et al* (2004, 2005) se desarrolló un algoritmo para la visualización de *MDEs*

mediante simplificación interactiva basado en estrategias *quad-tree* dirigidas por la curvatura local y la posición del observador. En estos algoritmos se trabajó con una metodología *top-down*, que transforma el campo de alturas original en un campo cuadrado de longitud de lado  $L/2^k$ , donde  $k$  es el nivel de refinamiento y  $L$  el tamaño del dominio. Luego, un algoritmo  $O(n \log n)$  efectúa los sucesivos refinamientos generando un árbol cuaternario restringido por niveles del cual pueden extraerse aproximaciones del terreno con resolución variable. El grado de detalle se determina imponiendo una cota superior a la acumulación de un indicador extensivo (*i.e.* proporcional a la superficie) de curvatura corregido con la distancia al observador. El criterio es que las regiones con más curvatura y más cercanas al observador requieren mayor nivel de detalle. De esta forma los lugares con baja curvatura pueden ser simplificados sin perder calidad. Una restricción adicional impone como máximo un nivel de subdivisión de diferencia entre regiones *quadtrees* adyacentes, lo cual garantiza la conformidad de la malla de la superficie (*i.e.* sin agujeros). La [figura 1](#) muestra una situación en la que se producen agujeros en una representación en perspectiva y cómo la restricción asegura la conformidad de la malla (Cifuentes *et al*, 2005).

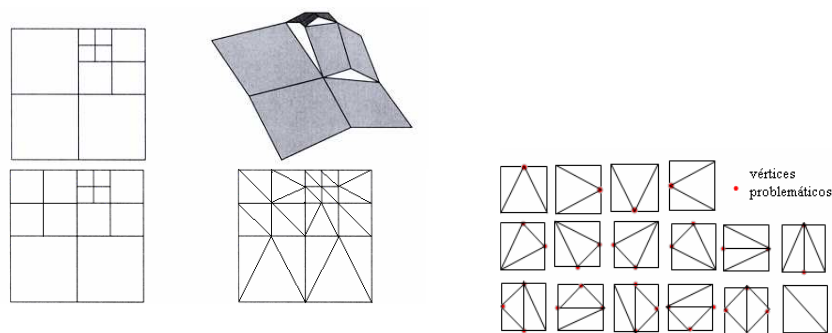


Figura 1 – Arriba: quadtree sin restringir y su representación en perspectiva. Abajo: quadtree restringido y triangulado según el template de la derecha.

En este trabajo se presenta una nueva estrategia de simplificación de mallas de MDE basadas en *quad-trees*. El método general parte de una triangulación de referencia que luego se zonifica mediante un indicador acumulado de curvatura, distancia al observador y campo visual. Los nodos terminales del *quad-tree* se triangulan en una de cinco posibles situaciones que son identificadas rápidamente gracias a la localización directa de regiones vecinas. Este primer paso da como resultado una triangulación base de la superficie. La representación visual en cada instante se actualiza dinámicamente conforme cambian los parámetros de visión asociados a un observador inmerso en la escena. El detalle conveniente en cada zona lo establece un nuevo indicador que combina la posición e inclinación de la vista del observador. De esta manera en cada instante se identifican las regiones que deben ser remalladas y luego se modifica la aproximación base agregando precisión al escenario en estas regiones (estrategia *bottom-up*). Los resultados obtenidos muestran una mejora de la velocidad de visualización en un sobrevuelo pasante sobre la superficie, comparando contra una estrategia de remallado dinámico completo de la escena.

## 2 ESTRATEGIA DE SIMPLIFICACIÓN *BOTTOM-UP*

La estrategia *bottom-up* para simplificación de mallas parte de un árbol cuaternario restringido del que pueden extraerse diversas aproximaciones. El nuevo algoritmo formulado construye la aproximación en  $O(n)$ , es decir  $n$  cotas definen la superficie y constituyen los nodos terminales del *quadtree*, a los cuales se asigna un indicador de curvatura aplicando un esquema centrado en diferencias finitas al campo escalar de alturas (Cifuentes *et al* 2005). Los restantes niveles del árbol se generan agrupando estos nodos terminales, por lo que cada nodo intermedio acumula las curvaturas de sus cuatro descendientes. La agrupación de nodos descendientes queda determinada por la rugosidad del terreno y la necesidad de mantener la cohesión de la malla (sin agujeros). Para ello, el acumulado de curvaturas de una región no debe superar un dado valor tolerado ni diferir en más de un nivel de distancia en el árbol con sus adyacentes (Cifuentes *et al* 2005). El árbol resultante de este proceso conforma la representación base con mínimo detalle.

El proceso de triangulación del árbol de nodos base es relativamente sencillo. Para cumplir con la restricción de diferencia máxima de un nivel, cada región (cuadrada) se puede triangular en dos triángulos o según las 4 posibilidades que se muestran en la [Figura 2](#). Los triángulos en gris oscuro corresponden a subdivisiones forzadas que mantienen la estructura restringida. El vértice ficticio del centro de la región agiliza el proceso de triangulación, independizando la subdivisión de un sector de la región sucesiva y del resto (no obstante, no introduce error en la representación debido al criterio de coplanaridad impuesto para la simplificación del modelo). Esta metodología de subdivisión es una novedad introducida en este trabajo, que se diferencia de la usada en Cifuentes *et al* (2005), donde es necesario verificar el nivel de subdivisión de los cuatro nodos adyacentes para triangular cada nodo. Finalmente, se construye la *triangulación base*, resultando una malla poligonal con la que se puede visualizar la superficie con la apariencia solicitada por el usuario minimizando la cantidad de datos a manipular.

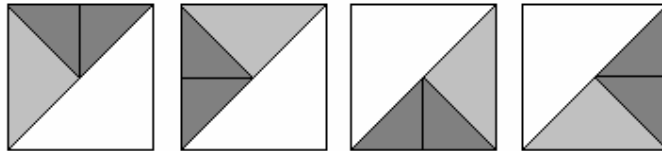


Figura 2- Template de triángulos para la generación de mallas conformes.

<b>Nodo</b>	<b>Nivel</b>	<b>Posición</b>
Ancestro:	$(h-1)$	$(x/2, y/2)$
Descendientes:	$(h+1)$	$(2x+0/1, 2y+0/1)$
Vecino al Oeste:	$h$	$(x-1, y)$
Este:	$h$	$(x+1, y)$
Norte:	$h$	$(x, y-1)$
Sur:	$h$	$(x, y+1)$

Tabla I – Localización de ancestros, descendientes y vecinos de un nodo en  $(x, y)$  de altura  $h$ .

De esta manera, se logra implementar la jerarquía *quadtree* usando una estructura sencilla de acceso y actualización, la cual consiste en un arreglo unidimensional de tamaño igual a la cantidad de niveles del árbol cuaternario a generar, cuyos elementos son matrices que organizan los nodos de cada nivel. La [Tabla I](#) ayuda en la localización de nodos ancestros, descendientes y vecinos ortogonalmente adyacentes de un nodo cuyo nivel es **h** y su posición en la matriz es **(x, y)**. En todos los casos el acceso a los nodos es directo produciendo la navegación eficiente del *quadtree*. Adicionalmente, los nodos conservan indicadores de área nula, de subdivisión forzosa y del nivel de detalle actual para mejorar la performance. El [Algoritmo I](#) describe un pseudocódigo del procedimiento propuesto.

```

para cada nivel del quadtree (de abajo hacia arriba)
    para cada nodo del nivel
        si (alguno de los adyacentes ó descendientes de nodo se divide)
            dividir nodo
        sino
            calcular K de nodo, suma curvaturas locales de los descendientes
        si (K > Tolerado)
            dividir nodo
        sino
            guardar K en nodo

```

Algoritmo I- Pseudocódigo *botton-up* de mallas topográficas

### 3 VISUALIZACIÓN ANISOTRÓPICA INTERACTIVA DE TOPOGRAFÍA

Como se mencionó en la introducción, el principal inconveniente de la visualización de modelos topográficos digitales es el volumen de información a manipular. Aunque la simplificación de la malla de representación reduce la cantidad de memoria, esto no alcanza para otorgarle el dinamismo necesario a la escena. Afortunadamente, en visualizaciones interactivas todavía es posible mejorar aún más el manejo de datos. Por ejemplo al efectuar travesías aéreas virtuales, resulta innecesario conservar y/o representar toda la información, ya que en cada momento sólo interesa refinar la parte del terreno que se encuentra en la dirección de desplazamiento del vehículo. Es decir, la visión desde un dado punto del terreno es anisotrópica.

Para tener en cuenta este último efecto, se parte de una malla base con un nivel de detalle mínimo (*i.e.* construida con una tolerancia alta de curvatura y distancia), la que será refinada anisotrópicamente durante la visualización dinámica. El indicador que se utiliza como métrica de la pertenencia de una región al campo de visión es:

$$e = \frac{\cos \theta}{d} \quad (1)$$

donde  $d$  es la distancia del observador al punto de prueba y  $\theta$  es el ángulo que forma la dirección del movimiento del observador y la dirección del vector que une la posición del observador con el centro de la región.

Para obtener la triangulación instantánea de la superficie, primero se determina la región

cuyo  $e$  es máximo. Esa región se refina con un detalle mayor de acuerdo a un criterio impuesto por el usuario, y a partir de allí se refina aplicando una estrategia *bottom-up*. Se encontraron buenos resultados con un refinamiento de dos niveles de árbol. El pseudo-código de este método es el siguiente:

*Triangulación anisotrópica = triangulación base;*  
*Refinar la región con  $e$  máximo en 2 niveles de detalle*  
*Para cada descendiente de esa región restringir triangulación anisotrópica*

El paso de restricción sirve para conformar la malla anisotrópica para que no tenga agujeros, para lo cual se actualiza el nodo que invoca y luego se autoinvoca con el ancestro y los adyacentes del ancestro; de esta forma, el quadtree generado de abajo hacia arriba, también es restringido. Este proceso recursivo de restricción finaliza con el hallazgo de un nodo ya restringido o cuando se alcanzó el primer nivel del árbol. Así, la triangulación base constituye el marco de referencia para la generación de esta triangulación anisotrópica subordinada. La representación interna de los datos posibilita que tal generación simultáneamente pueda restringirse y elaborarse en  $O(n)$ , donde  $n$  es el número de cotas que definen la superficie.

#### 4 RESULTADOS

El algoritmo presentado fue aplicado a un modelo digital de elevación del cañón del Colorado (16.764.930 de triángulos). La aplicación visualizadora se desarrolló íntegramente en C++ y la interfaz de usuario aprovecha las utilidades de la librería OpenGL (Woo *et al* 1997). La métrica de calidad de malla utilizada es la suma de las diferencias de cuadrados entre pares de cotas pertenecientes a la aproximación y la malla original cruda. Por otro lado, la eficiencia de la simplificación es medida con el porcentaje de triángulos que se incorporan a la *triangulación base* para agregar el detalle requerido en el campo de visión del observador.

A pesar de los importantes cambios de pendiente en toda su extensión, se han logrado excelentes visualizaciones con simplificaciones con menos del 1 % de los polígonos originales (ver [Figura 3](#)). Incrementando la tolerancia asociada al indicador de error, se estudiaron siete simplificaciones del modelo original: las *triangulaciones base* (en negro, ver [Figura 4](#)) compuestas por 147018, 35889, 6021, 2774, 1964, 1546 y 912 triángulos. Y, dejando el observador ubicado en una dada posición, se construyeron también las respectivas *triangulaciones anisotrópicas* (en gris, ver [Figura 4](#)) compuestas por 147178, 36046, 6227, 2866, 2071, 1753 y 1007 triángulos. Se puede ver que el mayor detallado en los alrededores del observador incrementa sólo un pequeño porcentaje la cantidad de polígonos de la aproximación base.

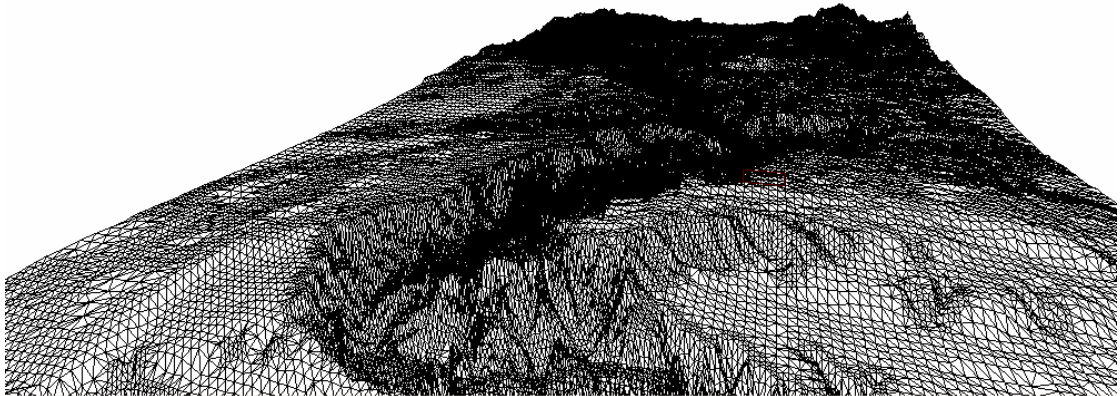


Figura 3- Cañón del Colorado, malla poligonal aproximada de 147018 triángulos.

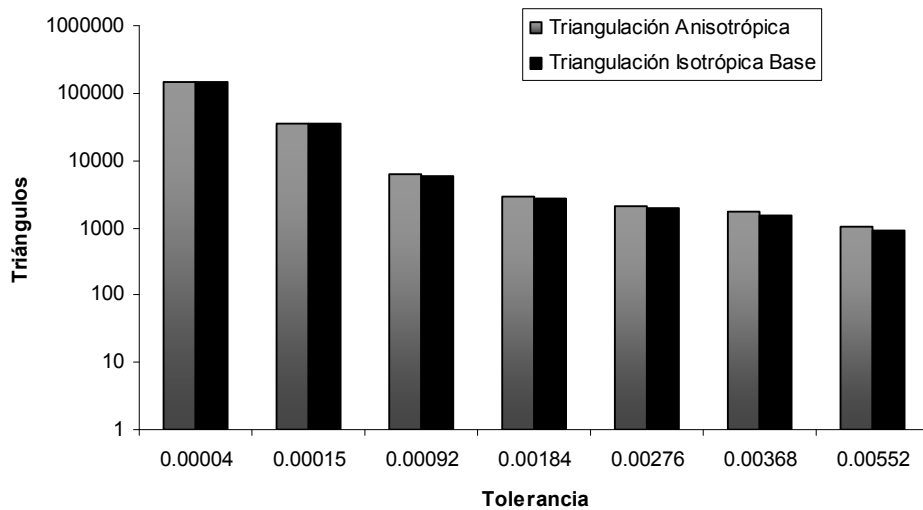


Figura 4- Cañón del Colorado: comparación entre simplificaciones y anisotrópica para distintas tolerancias

En la figura 4 se visualiza una aproximación anisotrópica del modelo digital de 16,8 millones de triángulos. Si bien la triangulación base se compone de 14681 triángulos, la vista sólo representa a 6970 triángulos logrando la conjunción de todos los algoritmos en una representación interactiva con 18 fps. Nótese que, la parte inferior de la captura representa el primer nivel de detalle agregado a la visualización, cuya precisión que puede aumentarse gradualmente con el dinamismo del observador hasta alcanzar el detalle máximo.



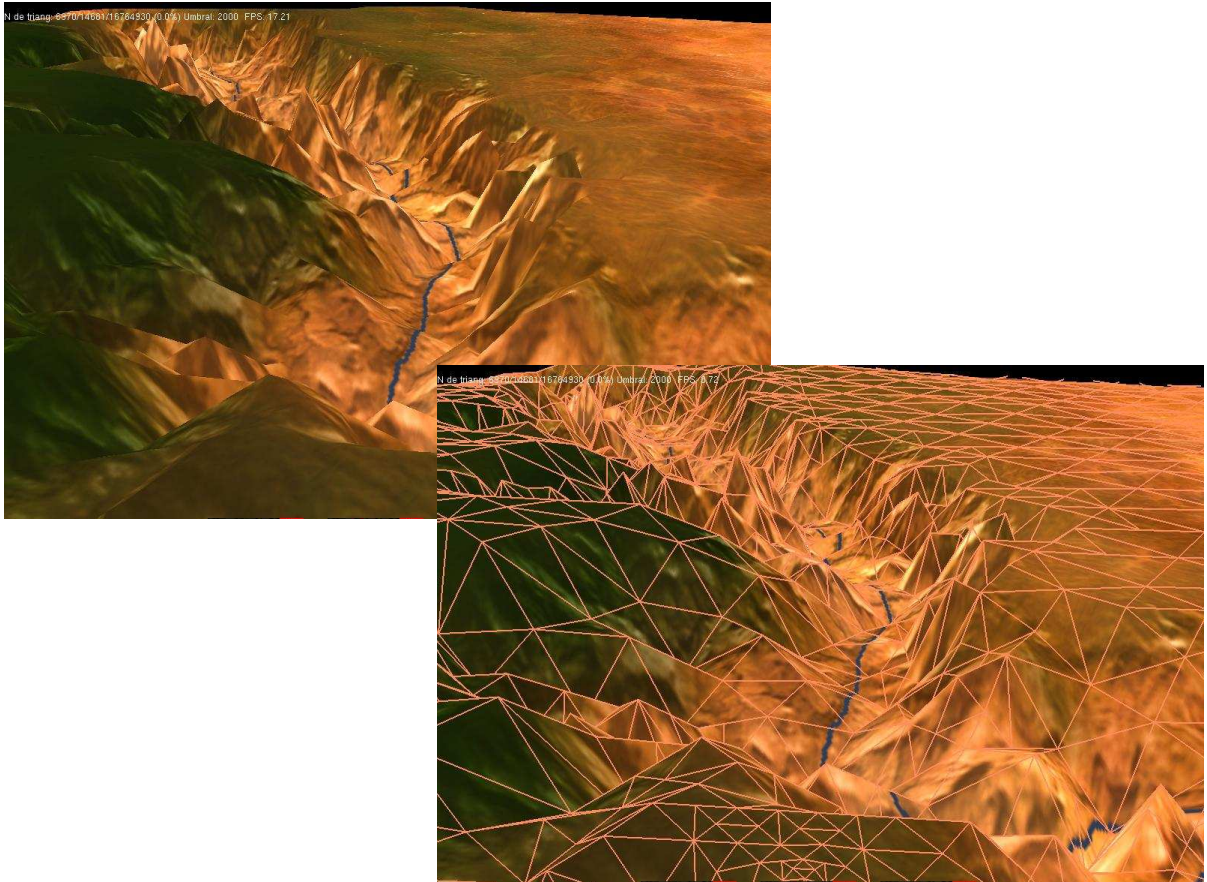


Figura 4 – Cañón del Colorado: malla poligonal anisotrópica con 14681 triángulos.

## 5 CONCLUSIONES

Se presentó un algoritmo *bottom-up*  $O(n)$  para la simplificación y visualización interactiva de modelos topográficos, con la cual se genera una jerarquía de mallas poligonales del mismo terreno con distinta resolución y evita la presencia de agujeros en la trama restringiendo la estructura. La navegación de la estructura en la búsqueda de las regiones vecinas, ancestros y descendientes es directa. El algoritmo de remallado mantiene la calidad visual tolerada por el usuario y detalla aún más las regiones dentro del campo de visión directo del observador. La selección automática de detalle se logra mediante un indicador que combina la posición y dirección de la vista. El algoritmo presentado simplifica notoriamente el volumen de datos a visualizar respecto de la grilla regular original.

## REFERENCIAS

- Abásolo, M.J., Blat, J., De Giusti, A., A Hierarchical Triangulation for Multiresolution Terrain Models. *Journal of Computer Science and Technology*, v.1, n.3, *Computers Science Research: State of the Art*, 2000.
- Balmelli, L., Kovacevic, J., Vetterli, M., Quadrees for embedded surface visualization: constraints and efficient data structures. *In Proceedings of IEEE International Conference on Image Processing*, 2:487-491, 1999.
- Cifuentes, M.V., Vénere, M.J., Clause, A., Un algoritmo para la simplificación poligonal de modelos topográficos digitales. *33° JAIIO, Jornadas Argentinas de Informática e Investigación Operativa*, 2004.
- Cifuentes, M.V., D'Amato, J.P., García Bauza, C., Vénere, M.J., Clause, A., Remallado interactivo como método para la navegación eficiente de modelos topográficos. *Mecánica Computacional*, 24:2385-2395, 2005.
- Cifuentes, M.V., D'Amato, J.P., García Bauza, C., Vénere, M.J., Lotito, P., Clause, A., Ray Casting para la definición de zonas de interés en simplificación topográfica. *Mecánica Computacional*, 25:1177-1186, 2006.
- Gross, M.H., Gatti, R., Staadt, O., Fast Multiresolution Surface, Meshing. *Proceedings of Visualization '95*, p. 135-142, 1995.
- Hoppe, H., View-Dependent Refinement of Progressive Meshes. *Computers & Graphics. Proceedings of SIGGRAPH'97*, p.189-198, 1997.
- Lindstrom, P., Pascucci, V., Visualization of Large Terrains Made Easy. *Proceedings of IEEE Visualization, San Diego, California*, p. 363-370, 2001.
- Pajarola, R., Overview of quadtree-based terrain triangulation and visualization. Technical Report, UCI-ICS-02-01, Information & Computer Science, University of California Irvine, ical Methods in Engineering, 2002.
- Rottger, S., Heidrich, W., Slusallek, P., and Seidel, H. P., Real-Time Generation of Continuous Levels of Detail for Height Fields. *V. Skala, editor, Proceedings of of the 6th International Conference in Central Europe on Computer Graphics and Visualization*, pages 315-322, 1998.
- Vénere, M.J., Cifuentes, M.V., D'Amato, J.P., García Bauza, C., Editor de escenarios para aplicaciones de Realidad Virtual. *34° JAIIO, Jornadas Argentinas de Informática e Investigación Operativa*, 2005.
- Woo, M., Neider, J., Davis, T., OpenGL Programming Guide: The Official Guide to Learning OpenGL. *Addison-Wesley Developers Press*, 1997.