

Desarrollo de algoritmos de teselación adaptativa en GPU

Proyecto Final de Carrera

Alumno: Fernando Nellmeldin

Director: Dr. Néstor Calvo

Facultad de Ingeniería y Ciencias Hídricas
Universidad Nacional del Litoral

12 de Septiembre de 2014

- 1 Introducción
- 2 Teselación
- 3 Métricas
- 4 Implementación
- 5 Comentarios finales

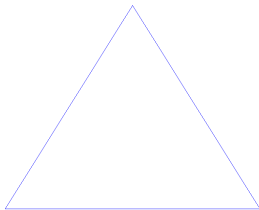
Representación de objetos 3d en computadoras



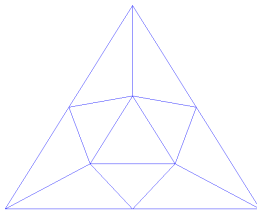
Modelo de personaje utilizado en videojuegos. *Fuente: Capcom Europe*

Teselación

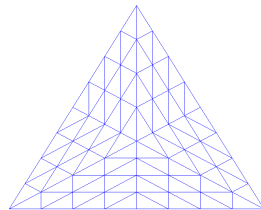
- ¿Qué es?
- ¿Para qué sirve?



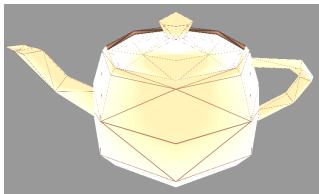
(a)



(b)



(c)



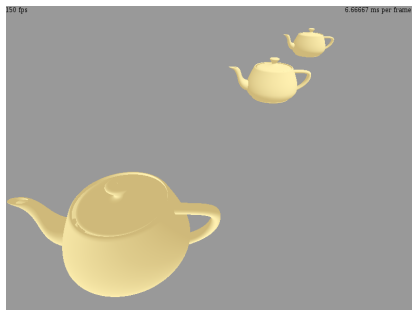
(d)



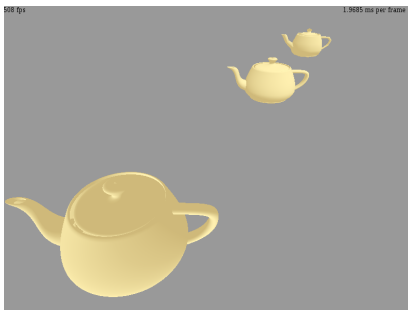
(e)

Adaptatividad

- ¿Para qué?
- ¿Cómo se logra?

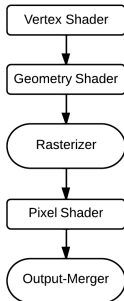


(a) Sin adaptatividad (6.7 ms).

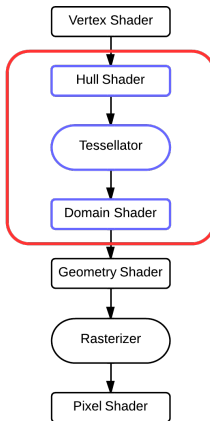


(b) Con adaptatividad (2.0 ms).

Pipeline DirectX 10
(2007)



Pipeline DirectX 11
(2009)



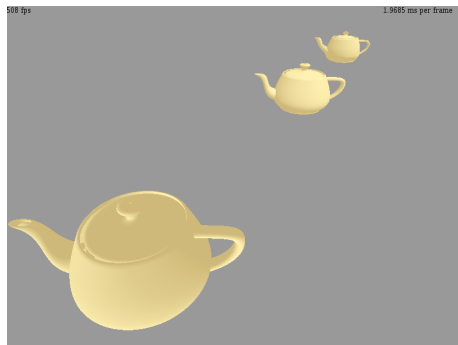
Cambios en el pipeline gráfico.

Ventajas y desventajas de teselación en GPU

- 😊 Paralelismo
- 😊 Menor uso de memoria principal
- 😊 Menor transferencia de datos
- 😊 Simplicidad

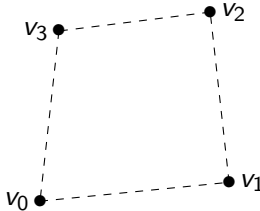
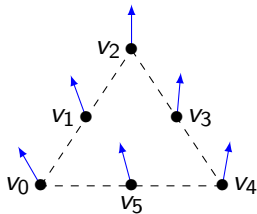
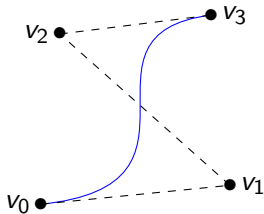
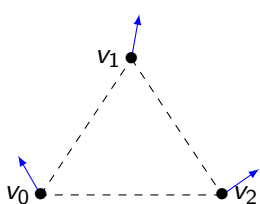
- 😞 Menor control
- 😞 Sólo en hardware y software moderno

- Investigar la tecnología de teselación en GPU.
- Diseñar métricas.
- Implementar modelos-ejemplo.
- **Controlar el nivel de detalle dinámicamente en GPU.**

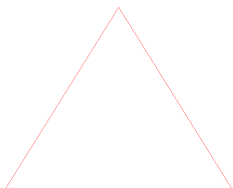


Definición

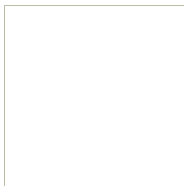
Conjunto ordenado de vértices y atributos (posición, normal, coordenadas de textura, etc.).



Tipos de subdivisión



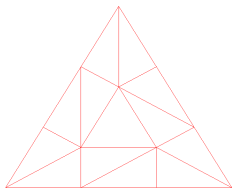
(a) Espacio triangular.



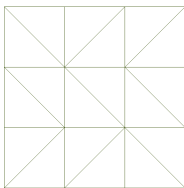
(c) Espacio cuadrado.



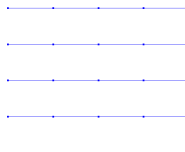
(e) Espacio cuadrado.



(b) Triángulos.

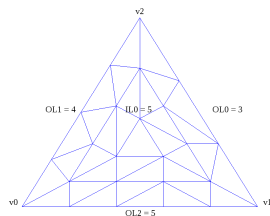


(d) Cuadriláteros.

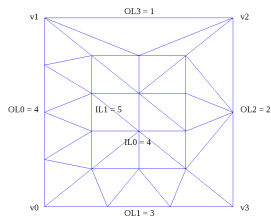


(f) Poligonales.

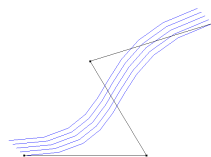
Niveles de teselación



(a) Triángulos.



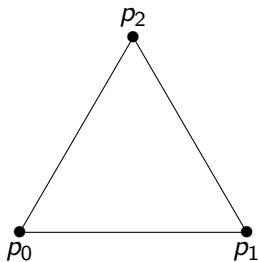
(b) Cuadriláteros.



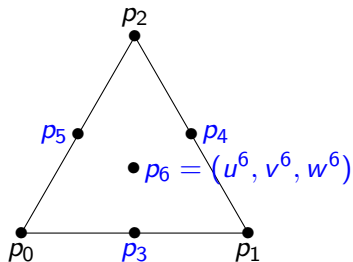
(c) Poligonales.

Teselación ejemplo de un patch de 3 vértices

Espacio paramétrico (generación de coordenadas u, v, w):



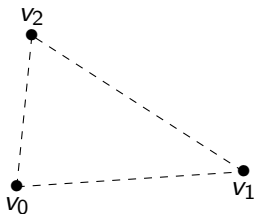
(a) Dominio inicial.



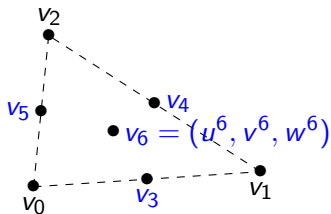
(b) Dominio subdividido.

Teselación ejemplo de un patch de 3 vértices (cont.)

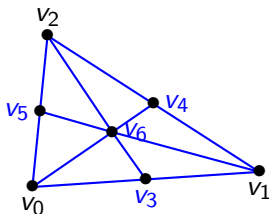
Espacio del modelo (interpolación utilizando las coordenadas u, v, w):



(a) Patch inicial.

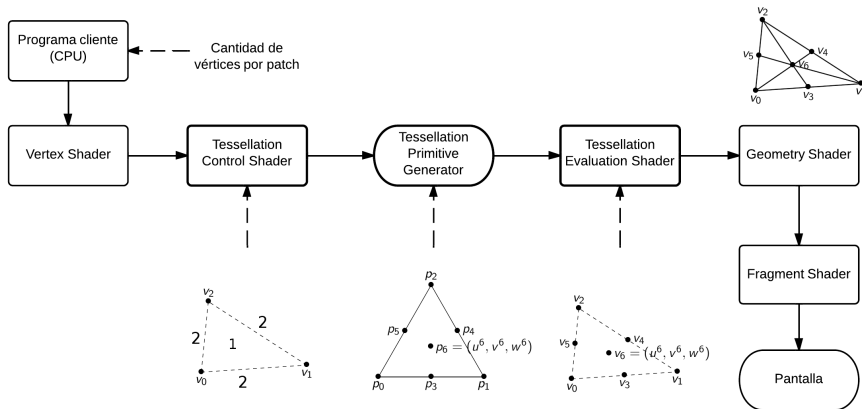


(b) Mapeo de la subdivisión.



(c) Patch teselado.

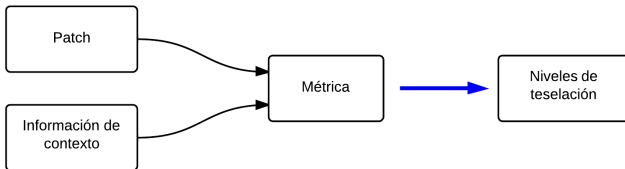
Pipeline en OpenGL y responsabilidades



Responsabilidades de cada etapa del pipeline gráfico.

Definición

Medida obtenida a partir de la información del patch y el contexto, que permite determinar los niveles de teselación a aplicar al patch.

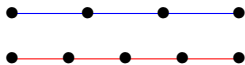


Clasificación

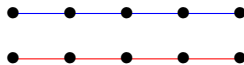
- Intrínsecas
- Extrínsecas

Características

- Simplicidad
- Poca información
- **Adaptatividad**
- Libre de grietas (uniones en T)



(a) Ejemplo de unión en T.



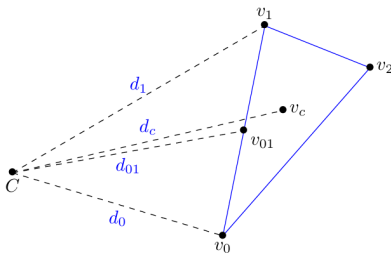
(b) Eliminación de la unión en T.

Métricas implementadas

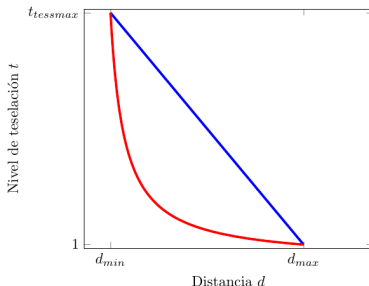
- Distancia a la cámara (extrínseca)
- Curvatura (intrínseca)
- Longitud de arista (extrínseca)
- Posición en silueta (extrínseca)

Distancias calculadas

- Por cada arista
 - Distancia a cada vértice.
 - Distancia al punto medio de la arista.
- Distancia al centroide del patch.



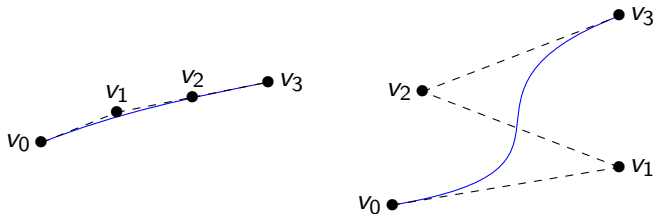
(a) Cálculo de las distintas distancias.



(b) Mapeo de distancia a nivel de teselación.

Noción de curvatura

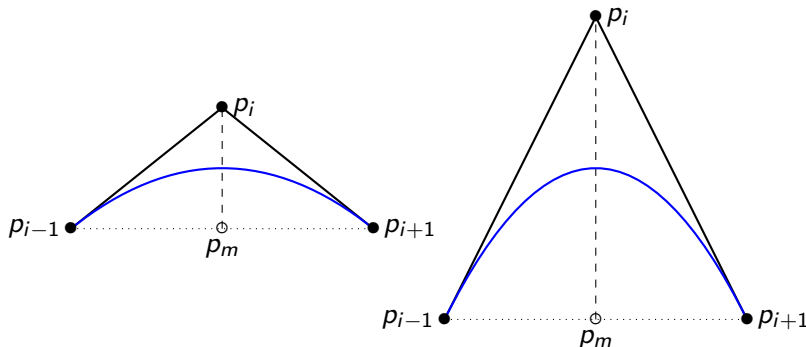
Desviación respecto a un plano.



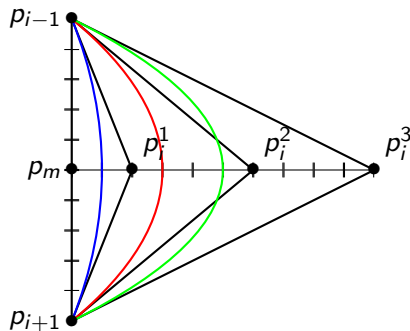
Distintos grados de curvatura.

Indicador de curvatura

$$\kappa = \frac{\|p_i - p_m\|}{\|p_{i-1} - p_{i+1}\|}$$

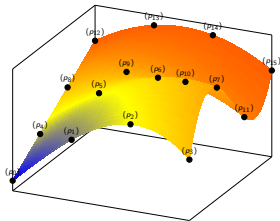


Variación de la curvatura al mover un único punto de control.

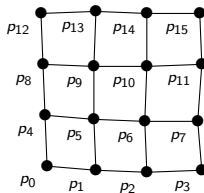


Curvas con distinta curvatura \Rightarrow Distintos niveles de teselación.

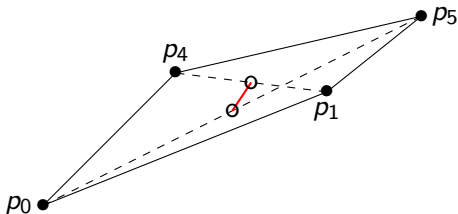
Extensión a superficies



(a) Superficie en el espacio.



(b) Malla de control.

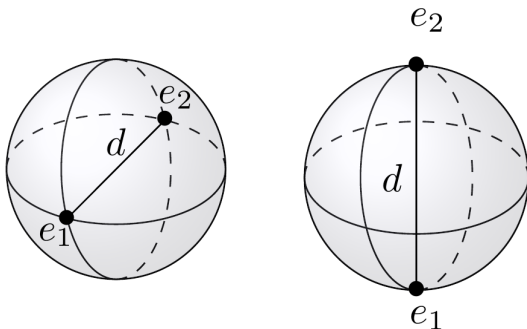


(c) Cálculo de curvaturas diagonales.

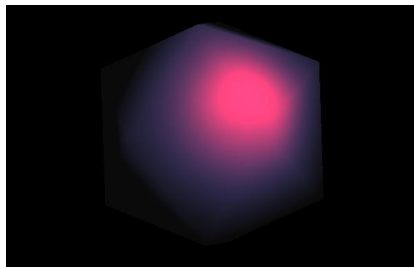
Métrica de longitud de arista

Opciones al calcular la proyección de una arista

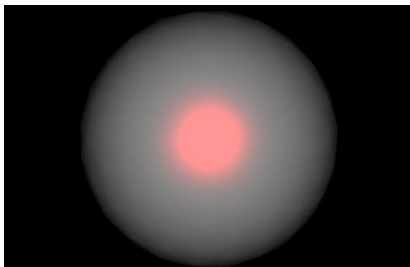
- Proyección directa.
- Proyección del diámetro de la esfera que la contiene.



Proyección de una arista como diámetro de la esfera que la contiene.



(a) Silueta recta.

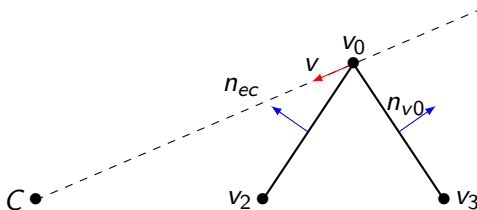


(b) Silueta curva.

Ilustración de los problemas de sombreado en la silueta.

Definición de arista silueta

Las normales de los dos triángulos de los que es frontera tienen orientaciones distintas.



Ejemplo de arista silueta.

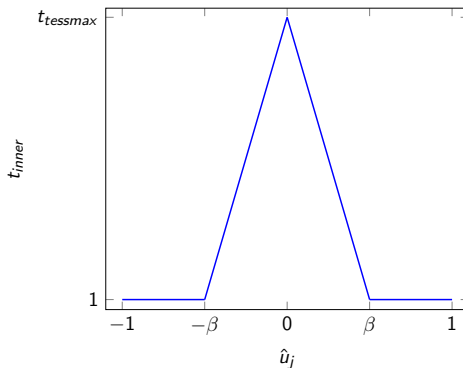
Niveles externos

$$u_j = (v \cdot n_{ec}) (v \cdot n_{vj})$$

$$t_j = \begin{cases} t_{tessmax}, & u_j < 0 \\ 1, & u_j \geq 0 \end{cases}$$

Nivel interno de teselación

$$t_{inner} = \begin{cases} -\frac{t_{tessmax}-1}{\beta} |\hat{u}_j| + t_{tessmax} & u_j < 0, |\hat{u}_j| \leq \beta, 0 < \beta \leq 1 \\ 1, & \text{en otro caso} \end{cases}$$



Función de mapeo de \hat{u}_j al nivel de teselación interno.

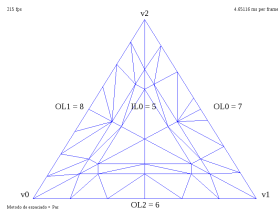
Tecnologías y bibliotecas utilizadas

- OpenGL 4.2 (2011).
- GLUT.
- Assimp.
- GLM.
- DevIL.
- FreeType.

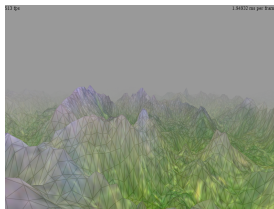
Características del software

- Selección de ejemplos.
- Diseño modular.
- Control de parámetros de teselación.
- Distintos modos de visualización (iluminación, wireframe, colores de teselación).

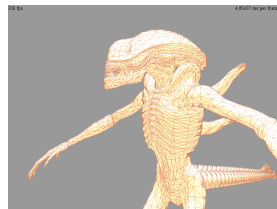
Modelos utilizados



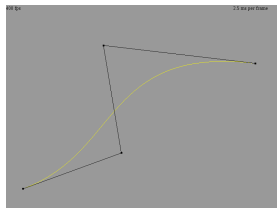
(a) TessDemo



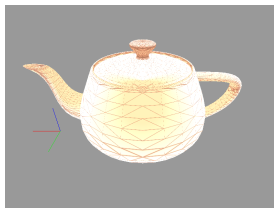
(b) Terreno



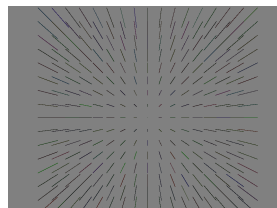
(c) Modelo de personaje



(d) Curva de Bézier



(e) Superficies de Bézier



(f) Campo vectorial

- TessDemo.
- Superficies con métrica de distancia.
- Curva con métrica de curvatura.
- Superficies con métrica de curvatura.
- Terreno con métrica de longitud de arista.
- *Phong Tessellation*.
- Personaje con métrica de silueta y *Phong Tessellation*.
- Terreno con métrica de distancia.
- Ventajas de la corrección diádica.

Conclusiones

- Conocimiento completo de la tecnología.
- Tecnología sencilla de utilizar.
- Teselación sin grietas (uniones en T).
- Necesidad de definir un límite en los niveles de teselación.
- Elección de métricas según modelo.

Trabajo futuro

- Teselación en superficies de subdivisión.
- Teselación en problemas de ingeniería (Método de Elementos Finitos).

