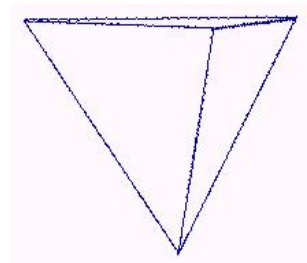
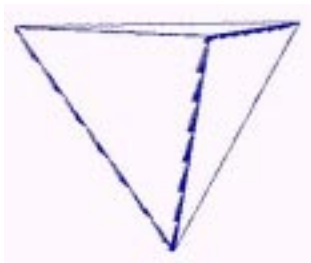
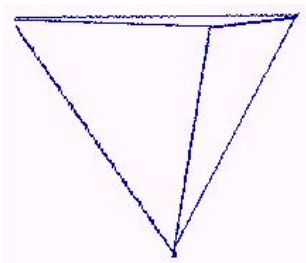


Rendering NPR: Aplicación de distintos estilos de trazos a siluetas de modelos 3D



González Adela., Zavala Alfredo

Proyecto: Sistemas Inteligentes para Scheduling y Control¹

Departamento de Informática

Universidad Nacional de San Luis (UNSL)

Ejército de los Andes 950

5700, San Luis, Argentina

e-mail: {adelag, azavala}@unsl.edu.ar

Resumen

Gran parte de la investigación en computación gráfica ha reconocido la potencia y utilidad del no fotorealismo. El rendering NPR básico es usado desde los sistemas CAD hasta en animaciones proveyendo de esta manera una herramienta alternativa para modelos 3D.

Este trabajo presenta un generador de imágenes no fotorealístico (render NPR) de modelos 3D el cual reproduce la silueta de los objetos en la escena con diversos estilos de trazos (strokes) de lápiz en una forma relativamente rápida. Se analizan métodos de generación de trazos.

Palabras claves: Computación Gráfica. Rendering no-fotorealístico. Modelos 3D. Stroke. Rendering.

¹ Director: PhD. Raúl Gallard. El grupo de investigación está soportado por la Univ. Nacional de San Luis. <http://www.lidic.unsl.edu.ar/>

1. Introducción

Gran parte de la investigación en computación gráfica[2] ha reconocido la potencia y utilidad del no fotorealismo [1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]. El rendering realístico es usualmente más costoso computacionalmente y algunas veces omite importantes sugerencias visuales en una escena.

Un rendering es una abstracción que favorece, preserva o aún enfatiza algunas cualidades mientras que suprime u omite otras que no están en el foco de atención. Gran parte de las actividades de rendering en computación gráfica tienen que ver con fotorealismo, es decir, tratan de emular una imagen que parece ser una fotografía de alta calidad. Este objetivo, si bien es útil y apropiado en varias aplicaciones no lo es en otras tantas[11, 12].

La investigación en rendering no fotorealístico (NPR) busca proveer métodos alternativos tanto para los modelos 3D como para imágenes de referencia. El rendering NPR básico es usado desde los sistemas CAD hasta en animaciones proveyendo de esta manera una herramienta nueva para modelos 3D.

Dentro del área de NPR el estudio de técnicas y métodos para resaltar contornos o bordes visibles (silueta) de un objeto[13] es un campo de gran interés.

Los dibujos en jaula de alambre (wire frame) suelen ser líneas de arte monocromático pero frecuentemente son muy detallados y duros de leer por lo que en algunas situaciones no es suficiente que el rendering este basado en líneas de arte blanco y negro. Al resaltar bordes de siluetas se puede realzar significativamente la comprensión de imágenes técnicas.

Algunos trabajos realizan una taxonomía de los estilos de líneas con una semántica asociada, ejemplos de tipos de líneas con geometría semántica: limite o borde, silueta, discontinuidad y líneas de control isoparamétricas, estas últimas no son líneas de contornos en el espacio del modelo pero ayudan a indicar curvatura. Las líneas pueden variar el grosor, la transparencia y el estilo (patrón sólido, punteado, etc.). Los atributos de las líneas dependen del tipo de línea (silueta, borde, etc.).

Este trabajo presenta un generador de imágenes no photorealístico de modelos 3D el cual reproduce la silueta de los objetos en la escena con diversos estilos de trazos (strokes) de lápiz en una forma relativamente rápida. Se analizan métodos de generación de trazos con varios estilos, algunos de los efectos logrados son carbonilla, lápiz flojo, lápiz a mano, bosquejo áspero.

2. Trabajos relacionados

Winkenbach y Salesin[4] representan los strokes o trazos generalmente como curvas paramétricas con desplazamientos para representar el grosor del trazo. Define un stroke S como un conjunto de segmentos de línea, un paso y una función característica. El paso $P(t):[0,1] \rightarrow \mathbf{R}^2$ resulta de usar una curva para aproximar los segmentos de línea donde pueden ser aplicadas diferentes funciones. Ellos usan curvas Bezier y B-Splines para representar los strokes.

Strassmann[14] simuló un pincel de cerdas como una colección de curvas suavizadas trazadas alrededor del paso de una curva

Markosian[1, 13] usó varias técnicas para generar strokes expresivos: dibujar las líneas de los polígonos directamente con leves realces que se logran variando el ancho de la línea o el color; generar trazos perturbados artísticamente de alta resolución al agregar desplazamientos a la poligonal la cual está parametrizada (fig. 3 y 4 muestran resultados de nuestra implementación) y strokes texturados que siguen a la poligonal.

Hertzmann[15] creó trazos de pincel curvados para aproximar una imagen de referencia. Modeló los trazos como una curva B-Spline cúbica antialiased, cada uno con grosor y color especificados; usando el gradiente de la imagen para guiar la ubicación de los trazos.

Sousa y Bouchanan[8] presentaron un render 3D de lápiz de grafito para contornos y sombreados incorporando técnicas de rendering usadas por artistas para ese modelo.

3. Creación del Render: Bordes de Silueta

Existen varios pasos involucrados en la creación de un render.

- Encontrar los bordes de siluetas visibles.
- Dibujar los bordes con un trazo artístico.
 - Con trazo parametrizado.
 - Con trazo aproximado con una curva.

En el rendering de siluetas sólo se dibujan los borde de un objeto tal cual son vistos desde un punto de vista particular.

Un borde *silueta* es definido como: *un borde adyacente a un polígono front-facing y a uno back-facing.*

Un polígono es front facing, si el producto interior de su normal exterior y un vector desde un punto del polígono a la posición del observador es positivo, de otro modo el polígono es back facing.

Un borde es parte de la silueta si es borde adyacente a un polígono front facing y back facing. Un borde de silueta es front-facing si su cara adyacente mas cercana a la cámara es front-facing. De otro modo es back-facing (Lee Markosian[1])

Existen varios algoritmos para encontrar la silueta entre ellos:

- ✓ *Algoritmo de fuerza bruta o exhaustivo:* Chequea cada borde del modelo y registra aquellos que son bordes de silueta. Si bien es simple posee como desventaja el chequeo de todos los bordes del modelo cada vez que cambia el punto de vista.
- ✓ *Algoritmo de detección de silueta en forma aleatoria:* Los bordes de siluetas tienden a estar organizados a lo largo conectados en grupos o cadenas. Este método basándose en la idea anterior, analiza sólo una pequeña fracción de los bordes del modelo buscando la silueta. Cuando uno es

encontrado comienza a procesar el borde adyacente en el punto final del corriente, repite este proceso hasta que completa la cadena. Este algoritmo es rápido.

3.1 Implementación de la silueta

Para un agrupamiento poligonal, las siluetas son exactamente el conjunto de bordes agrupados que conectan un polígono back-facing con uno front-facing. Los métodos usuales para computarla es iterar sobre cada borde y procesar las normales de las caras adyacentes. Esta repetición debe realizarse cada vez que la posición de la cámara cambia.

Las fig. 1 a y b muestran el rendering en wire frame y la silueta de un objeto (teapot).

En nuestra implementación usamos una estructura de datos que es muy común en computación gráfica y está organizada jerárquicamente: *vértices*, *polígonos*, *superficies*, *objetos*. La escena está formada por objetos, los objetos están formados por superficies, estas por polígonos y estos últimos por vértices, se almacena también la normal de cada polígono.

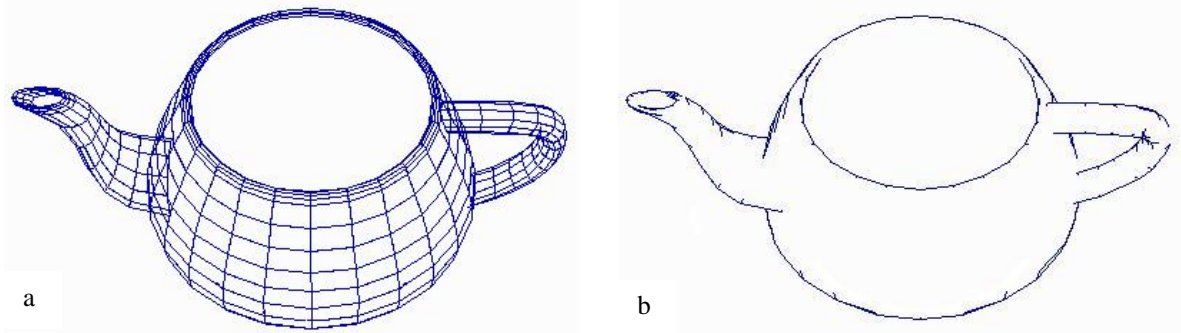


Fig. 1. Teapot: a) en jaula de alambre, b) Silueta del teapot

4. Creación del render: Dibujo de Trazos con estilo artístico

Podemos decir que un stroke (trazo) es una línea poligonal describiendo su paso central más información acerca del estilo a usar en el dibujo del trazo (este método es similar a los skeletal strokes de Hsu[16])

4.1 Stroke artísticos

Los strokes artísticos son definidos en el espacio de la imagen y la visibilidad de la silueta que ellos representan ya está asegurada. Con el uso de los strokes artísticos se simulan varios estilos de dibujo/pintura.

Un stroke puede ser implementado de varias formas. Dada una lista de pares de vértices podemos generar un trazo que pase a través de cada vértice creando un paso continuo. Es posible variar el ancho de la línea o perturbarla de diversas formas. A continuación mencionamos algunos métodos.

4.2 Implementación

Todos los bordes que pertenecen a la silueta encontrados en el punto 3 son individualmente procesados. Cada uno de éstos es tratado de acuerdo al método elegido por el usuario, ya sea como estilos logrados a través de la perturbación de una línea parametrizada o como aproximación de un paso con una curva Bezier o B-Spline.

4.2.1 Generación de una parametrización para dibujar los strokes

Cada borde es parametrizado como una línea que puede ser alterada en cada punto a través de desplazamientos. Con esto se pretende simular un bosquejo con una apariencia realizada por un humano.

La parametrización de un segmento v_1, v_2 es

$$pixel(t) = v_1 + t.(v_2 - v_1) \quad 0 \leq t \leq 1$$

Con los bordes de la silueta parametrizada podemos parametrizar la tangente y la normal de un borde. Tanto la tangente como la normal son usadas en conjunto con la función de brush como vectores de desplazamientos para el punto a dibujar.

La ecuación usada es la propuesta por Markosian[1]:

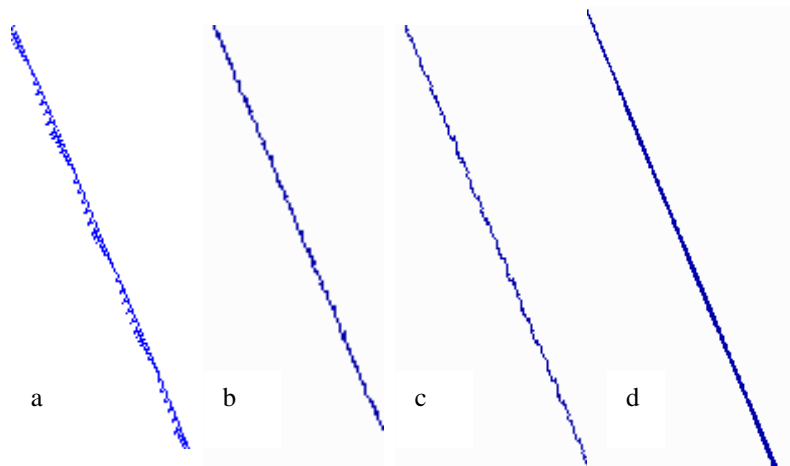
$$brush(t) = pixel(t) + v_x(t).pixel'(t) + v_y.n(t)$$

Donde

$brush(t)$	el valor del trazo del lápiz en t
$pixel(t)$	coordenadas del punto regular
$v_x(t)$	valor en x de la función del lápiz
$pixel'(t)$	tangente del borde en t
$v_y(t)$	valor en y de la función del lápiz
$n(t)$	normal del borde en t

El uso del vector de desplazamiento permite a $brush(t)$ realizar una vuelta a atrás sobre sí mismo o formar una iteración. Usando las bases tangente/normal se permite patrones de perturbación para seguir la curvatura de la silueta. Las variaciones sobre un pequeño número de la clase fundamental de stroke produce una amplia variedad de **estilos**.

Una función de pincel es una función que describe el sentido del lápiz a un nivel de detalle muy bajo. La función del lápiz para un trazo de:



a) *Carbonilla*: es una curva suavizada de alta frecuencia. La razón de esto es que puede ser visto como el dibujo de una escena en carbonilla sobre papel con lápiz.

b) *A mano*: ruido de baja magnitud aplicado a lo largo de la normal y la tangente.

c) *Lápiz flojo*: es una curva parabólica de baja frecuencia.

d) *Lápiz áspero*: ruido de alta magnitud aplicado a lo largo de la tangente.

Fig 2 Trazo: a) Carbonilla b) A mano c) Lápiz flojo d) Lápiz áspero

Las figuras 3 y 4 muestran la aplicación de este método a distintos objetos.

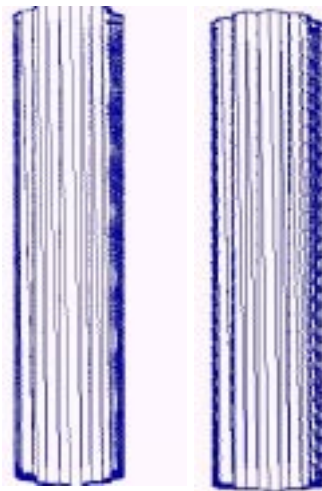


Fig. 3 Columna a) Carbonilla b) Lápiz flojo

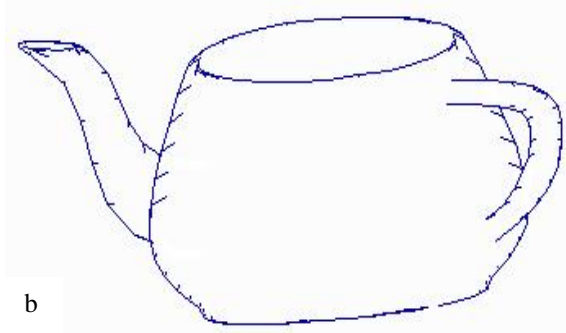
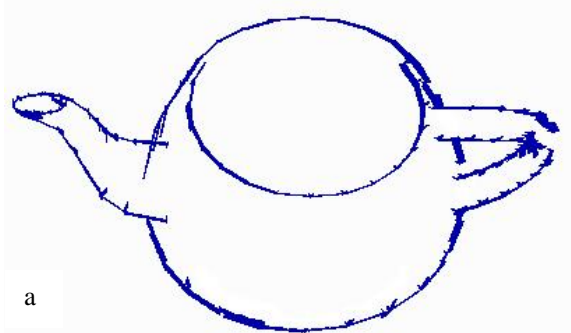


Fig. 4 Teapot: a) Silueta con trazos en bosquejo áspero, b) Silueta con trazos de dibujo a mano

4.2.2 Usando curvas paramétricas.

En este método un trazo consiste de un conjunto de segmentos de línea y un paso. El stroke resulta de usar una curva Bezier o B-Splines para aproximar los segmentos de línea donde diferentes funciones pueden ser aplicadas. Ver figura 5.

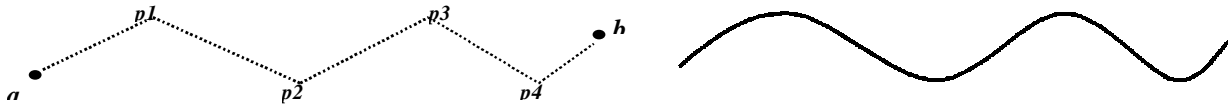


Fig. 5: Segmentos de línea y paso

Método:

- Se encuentran los puntos de control para la curva paramétrica. En este caso fueron definidos a distancias irregulares entre sí y perpendiculares al segmento a perturbar. Los mismos se encuentran alejados entre un 4% y un 10% de la longitud del segmento original. El sentido es aleatorio. La figura 6 muestra con un círculo sólido los puntos de control hallados por el algoritmo.

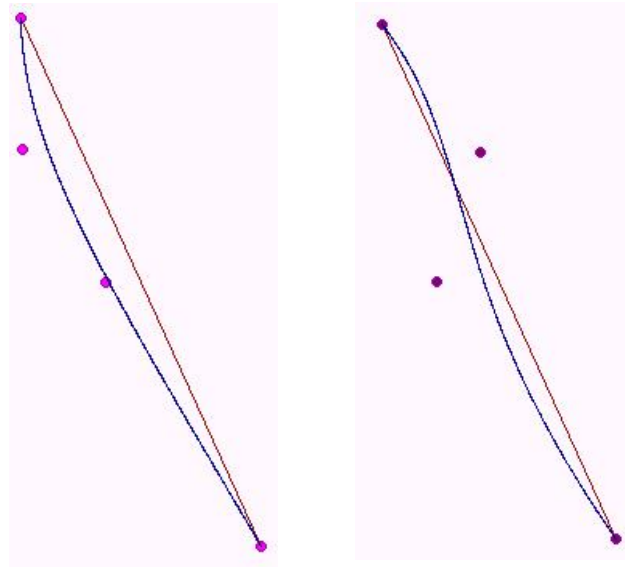


Fig. 6: Trazo con curvas Bezier

- Se puede alterar la curva paramétrica aplicando funciones de onda periódica (tal cual lo propone Sousa[8]) para simular los movimientos de la mano al modular aleatoriamente la curva que define el paso.

En la figura 7 se reconstruye el paso con una curva Bezier con 4 puntos de control.

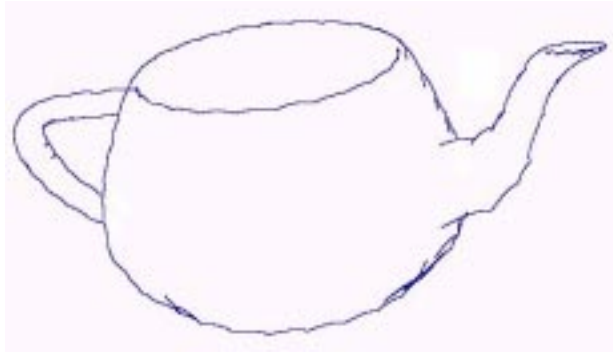


Fig. 7: Teapot

4.2.3 Perturbando la línea con ruido

Una técnica consiste en agregar distorsiones pseudo-aleatorias a los segmentos de línea para darle un look dentado. Este es un método demasiado duro y artificial tal cual se puede apreciar en la figura 8. Como estrategia para generar un estilo de dibujo se puede incorporar la *función de ruido estocástico de Perlin* de una dimensión para lograr perturbaciones con apariencia de dibujo a mano no repetitivo.

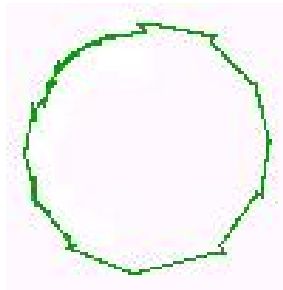


Fig. 8: Bola con ruido

El siguiente código perturba una línea con ruido de una manera muy simple.

```
procedure dibujeline(x1, y1, x2, y2);
  Ab <- 0.08;
  Paso <- 1/10;
  Long_line <- sqrt(sqr(x2-x1)+sqr(y2-y1));
  alfa <- arctan2(x2-x1, y2-y1);
  vertice <- (x1, y1);
  for (a=Paso ; a < 1; a++)
    n <- Ruido;
    x_nuevo <- x1 + a*(x2-x1) + n* Ab*sin(alfa)* long_line;
    y_nuevo <- y1 + a*(y2-y1) + n* Ab*cos(alfa)* long_line;
    vertice <- (x_nuevo,y_nuevo);
  end_for;
  vertice <- (x2, y2)
end_procedure;
```


5. Conclusiones y trabajo futuro

Los generadores NPR son muy interesantes dado que con ellos se pueden obtener resultados agradables. La investigación dentro de este campo y en particular en lo que respecta a la obtención de la silueta de un objeto y su posterior rendering usando alguna técnica, la cual puede incluir la de artista, ha captado mucha atención.

Presentamos un generador de imágenes no photorealístico de modelos 3D el cual reproduce la silueta de los objetos en la escena con diversos estilos de trazos (strokes) de lápiz en una forma relativamente rápida. La generación de los strokes se hizo usando varios estilos los cuales fueron analizados y se presentan algunos efectos logrados tales como carbonilla, lápiz flojo, lápiz a mano, bosquejo áspero.

Si bien son buenos los resultados obtenidos, el sistema podría mejorarse en el algoritmo de obtención de la silueta para poder aplicarse a animaciones así como en la búsqueda de nuevos métodos para encontrar estilos. Por otro lado, éstos métodos podrían ser aplicados a imágenes de referencia. Así también, es posible extender la generación de strokes para tono y textura.

Por último, éstos métodos pueden ser aplicados a ambientes de juegos o de realidad virtual.

6. Bibliografía:

1. Lee Markosian, Kowalski Michael A., Trychin Samuel J., et. al, *Real-Time Nonphotorealistic Rendering*, Brown Univeristy, SIGGRAPH'97
2. James D. Foley and Andries van Dam and Steven K. Feiner and John F. Hughes. *Computer Graphics, Principles and Practice*, Second Edition. Addison-Wesley , 1990, Reading, Massachusetts, Overview of research to date, sig-11-1994,
3. Michael P. Salisbury and Sean E. Anderson and Ronen Barzel and David H. Salesin, *Interactive Pen-And-Ink Illustration*, Proceedings of SIGGRAPH '94, 1994, july, pp. 101-108.
4. Geroge Winkwnback and David H. Salesin, *Computer-Generated pen-and-ink Illustration*. Proceedings of SIGGRAPH '94, 1994, july, pp. 91-100.
5. David Small, *Simulating watercolor by modeling diffusion, pigment, and paper fibers*. Proceeding of SPIE, Image Handling and Reproduction Systems Integration, vol. 1460, 1991, august, pp. 140-146.
6. Cassidy. J. Curtis, Sean E. Anderson, Joshua. E. Seims, Kurt W. Fleischer, and David. H. Salesin, *Computer-Generated Watercolor*. Proceeding of SIGGRAPH '97, 1997, august, pp. 421-430.
7. Mario C. Sousa and John W. Buchanan, *Observational model of blenders and erasers in computer generated pencil rendering*. Proceeding of Graphics Interface '99 1999, june, pp. 157-166.
8. Mario C. Sousa and John W. Buchanan, *Computer-generated graphite pencil rendering of 3d polygonal models*. Proceeding of Eurographics, 1999, september, pp. 195-207.
9. Mario C. Sousa and John W. Buchanan, *Observational models of graphite pencil drawing materials for non-photorealistic rendering*.

10. Julie Dorsey , Hans K. Pedersen and Pat Hanrahan, *Flow and Changes in Appearance*. Proceeding of SIGGRAPH '96, 1996, pp. 411-420.
11. John Lansdown and Simon Schoefield, *Expresive rendering: A review of nonphotorealistic techniques*. IEEE Computer Graphics and Applications, 1995, may pp.29-37.
12. B. Gooch, P. Sloan, A. Gooch, P. Shirley, R. Reisenfeld, *Interactiver Technical Illustration*, In Proc. of ACM Symposium on Interactive 3D Graphics, April 1999.
13. J. D. Northrup, Lee Markosian, *Artistic Silhouettes: A Hybrid Approach*, In Proc. of NPAR 2000.
14. S. Strassman, *Hairy Brushes*, In Proceeding of SIGGRAPH '86, 1986, august, pp. 225-232.
15. A. Hertzmann, *Painterly Rendering with Curved Brush Strokes of Multiple Sizes*, In Proc of SIGGRAPH 98, 1998, July, pp. 453-460.
16. Siu Chi Hsu and Irene H. H. Lee. *Drawing and Animation Using Skeletal Strokes*. Proceedings of SIGGRAPH 94, July 1994