

SA3D, um Sistema de Animação para Ambiente de Microcomputador

Carlos Gerardo de São Paulo *

José Miguel Salles Dias *,**

Jorge Vitória *

Manuel Gamito *

Pedro Faria Lopes *, **

Mário Rui Gomes *, ***

* INESC, Instituto de Engenharia de Sistemas e Computadores

R. Alves Redol, 9, 1000, Lisboa, Fax 351 1 525843

** ISCTE, Instituto Superior de Ciências do Trabalho e da Empresa

Av. das Forças Armadas, Edifício ISCTE 1600 Lisboa, Fax 351 1 7935300

*** IST, Instituto Superior Técnico, Av. Rovisco Pais, 1000, Lisboa

Sumário

Descreve-se a arquitectura do Sistema de Animação 3D (SA3D), um Sistema de Animação Modelada por Computador desenvolvido no projecto CAD/CAM no INESC, sob o ponto de vista da funcionalidade dos diferentes módulos constituintes : Modelação Geométrica 3D, Síntese de Movimento (Animação), Síntese de Imagem Realista e Edição e Digitalização 2D. Referem-se ainda alguns resultados obtidos com o Sistema, bem como a utilização do Sistema Gráfico CGI-INESC no desenvolvimento do módulo de edição 2D.

Palavras Chave : Animação por Computador, Modelação Geométrica 3D, Edição Gráfica 2D, Aproximação e Interpolação Spline, CGI-INESC

1. Introdução

No âmbito do projecto CAD/CAM do Instituto de Engenharia de Sistemas e Computadores, tem sido seguida uma linha de investigação no domínio da Animação por Computador [Lopes88b].

De facto desde Março de 1987, altura em que se finalizou o desenvolvimento do primeiro sistema de animação tridimensional, o ANIMED [Lopes87], consolidaram-se um conjunto de conceitos e tecnologias relacionados com esta área científica, incluindo Modelação Geométrica 3D, Interfaces Homem-Máquina Gráficas, Edição Gráfica 2D e 3D, Síntese de Movimento com aproximação cinemática, por interpolação de forma ou uma mistura destas duas aproximações, e Síntese de Imagem com os modelos e métodos : "wire-frame", constante, Gouraud, Phong e Ray-Tracing.

No seguimento desta estratégia, o INESC estabeleceu, em finais de 1988, um acordo com Carlos Gerardo de São Paulo para a realização de um novo Sistema de Animação para ambiente PC, capitalizando uma larga experiência, deste último, de mais de 5 anos de trabalho em Animação por Computador, quer ao nível de investigação e desenvolvimento, quer de produção.

O Sistema de Animação que se apresenta neste artigo, potencia assim uma larga experiência de Carlos Gerardo de São Paulo, o arquitecto do Sistema, com a experiência de 3 anos de investigação realizada no INESC, em Animação por Computador.

O Sistema SA3D foi desenvolvido em Linguagem C [Kernighan78] tendo em vista as limitações de um ambiente PC AT com sistema Operativo MS-DOS. Utilizou-se como suporte gráfico a placa Vista Videographics da AT&T, sendo a transportabilidade do sistema garantida através de especificações e desenvolvimento de um pequeno número de funções de baixo nível. Assim apenas foram desenvolvidas em "assembly", uma rotina que inicializa a placa com as normas video PAL GB, PAL M ou NTSC, uma outra que escreve um vector no "frame buffer", com uma cor (definida em RGB) e aplicando uma função lógica (cópia ou xor) e uma rotina que apaga todo o "frame buffer". Sobre esta rotinas foi desenvolvido todo o interface de entrada/saída gráfica do SA3D, nomeadamente um "pipeline" de visualização 3D e um "device driver" para o CGI-INESC.

As especificações do SA3D prevêm que as limitações quanto ao número de elementos suportados, por ex. numa malha de polígonos, são impostas pela dimensão da memória secundária (de disco) e não pela memória primária, sendo desse modo contornadas as limitações do Sistema Operativo MS-DOS.

Prevê-se que o sistema SA3D evolua para um sistema comercial cobrindo uma faixa de mercado com pequena capacidade de investimento, mas com grande dinâmica e espirito criativo.

2. Arquitectura do Sistema SA3D

O Sistema SA3D, um sistema de Animação Modelada por Computador [M. Thalmann85], no qual uma realidade complexa tridimensional é representada e animada nas estruturas internas de um sistema computacional, utiliza uma aproximação de especificação da animação que se baseia na aproximação/interpolação de parâmetros da cinemática do movimento, como o sejam ângulos de rotação, factores de escala, valores de translacção ou parâmetros de câmaras virtuais e fontes de luz [M. Thalmann85].

O sistema é composto, do ponto de vista da modelação, por um editor gráfico 2D, o ED2D, e por construtores geométricos procedimentais 3D. Quando o utilizador/animador pretende modelar uma série de objectos 3D, tem de invocar, ao nível do sistema operativo, um conjunto de construtores geométricos procedimentais, podendo visualizar rapidamente o resultado numa projecção ortogonal, através de um modelo de arame ou utilizando outros modelos de síntese de imagem ("Flat" e "Phong"). Quando pretende estabelecer uma animação hierárquica dos objectos 3D criados, deve escrever um guião de animação, definindo actores e subactores a partir dos mesmos [Lopes88b].

Nesse guião, utilizando as primitivas de animação do SA3D, o utilizador especifica procedimentalmente a hierárquia entre actores, estabelece as transformações, para cada quadro da animação, do espaço local dos actores para o espaço do mundo e manipula um número ilimitado de câmaras virtuais e um número finito de fontes de luz, podendo quando o desejar comutar entre elas. O guião é escrito em Linguagem C, devendo ser compilado e ligado às bibliotecas do SA3D de modo a ser produzido um ficheiro executável que corresponde ao filme criado. Quando tal ficheiro é executado produz-se, para cada quadro, um ficheiro com a descrição da imagem 3D calculada e projectada em 2D (imagem sintetizada). Este ficheiro imagem é visualizado no écran gráfico mediante um comando SA3D.

A arquitectura do Sistema SA3D está descrita no diagrama seguinte (Fig. 1) :

4. Modelação Geométrica 3D

O sistema SA3D suporta a descrição de modelos geométricos tridimensionais, sob a forma de malhas de polígonos [Foley84]. Cada polígono possui a seguinte descrição :

- lista vértices (com as coordenadas dos pontos cartesianos e coordenadas dos vectores normais nos pontos) percorridos no sentido anti-horário.
- vector normal ao polígono
- cor, RGB do polígono
- índice de uma tabela de materiais (mármore, cobre, plástico, etc.)

A criação de modelos 3D é feita à custa de construtores geométricos procedimentais incluindo construtores de : polígonos regulares convexos, prismas regulares, pirâmides regulares, esferas, superfícies de revolução, malhas de polígonos criadas pela propagação de uma curva ao longo de um caminho.

Polígonos regulares convexos

Dado o nome, o raio da circunferência que o inscreve e o número de lados, é gerado um polígono regular assente no plano XZ e com o centro na origem. O primeiro vértice gerado fica sobre o semi-eixo positivo dos ZZ.

Comando :

```
$ poligono outfile raio n_lados
```

```
$ poligono polig.m 100 4
```

Prismas regulares

Dado o nome, o raio da base, o raio do topo, a altura e o número de lados, são gerados os polígonos correspondentes às faces laterais de um prisma de base e topo regulares. A base fica assente no plano XZ e centrada na origem. O eixo do prisma é colinear com o eixo dos YY e o topo do prisma situa-se no seu semi-eixo positivo. A primeira aresta gerada fica assente no plano XY. Os polígonos gerados são guardados num ficheiro cujo nome é dado como primeiro argumento.

Comando :

```
$ prisma outfile raio_base raio_topo altura n_lados
```

```
$ prisma pris.m 20 10 50 6
```


Propagação de uma curva ao longo de um caminho

É possível a definição de uma malha de polígonos, propagando uma secção curva ao longo de um caminho. A secção é descrita no plano XY à custa de um vector de controle bidimensional, *secção.x* e *secção.y*, podendo assim corresponder a um segmento de recta, uma secção linear ou uma secção curva aberta ou fechada, dependendo tal facto do código do algoritmo utilizado no vector de controle.

O caminho de propagação da secção corresponde igualmente a uma curva descrita no plano XY. Essa curva é agora controlada pelo vector *caminho.x* e *caminho.y*. Definem-se igualmente o número de pontos avaliados na secção (resolução paramétrica u) e o número de vezes em que a secção é propagada ao longo do caminho (resolução paramétrica v). A propagação é realizada de tal forma que a secção se encontra sempre disposta ortogonalmente em relação à curva de propagação. A malha de polígonos resultante é guardada num ficheiro de saída *modelo.m*, sendo visualizada em projecção ortogonal à medida que é construída.

Por outro lado à medida que a secção se propaga pode-se opcionalmente definir uma transformação de escala, que se aplica à própria secção. Essa transformação define-se através de um vector de controle unidimensional.

A generalidade deste comando permite por ex. propagar uma spline fechada ao longo de um segmento de recta, obtendo-se previsivelmente um tubo. Mas escalando-se a secção por intermédio de uma outra spline ao longo da propagação, obtém-se um cálice (ver Fig. 3).

Se a secção corresponde a um segmento de recta a propagação corresponde a uma extrusão simples (ver Fig. 4). Por outro lado, se a secção e o caminho correspondem a splines fechadas obtêm-se, na malha de polígonos resultante, toros.

Comando :

```
$propag secção.x secção.y resol_u caminho.x caminho.y resol_v modelo.m escala
```

```
$ propag anel.x anel.y 8 linha.x linha.y 20 calice.m escala
```

Refira-se que as curvas associadas aos vectores de controle, que podem ser definidos com um vulgar editor de texto, ou, com mais precisão, utilizando o editor ED2D (ver secção 3), podem ser rapidamente visualizadas no écran gráfico com os comandos **plot** (vector bidimensional) e **spline** (vector unidimensional). Ex :

```
$ plot anel.x anel.y -aval 20          20 avaliações
```

```
$ spline escala -aval 30             30 avaliações
```

Os modelos criados podem ser visualizados graficamente, em projecção ortogonal e modelo de arame, por utilização do comando **projec** :

```
$ projec calice.m
```




Fig. 2: Da esquerda para a direita e de cima para baixo temos quatro curvas todas com o mesmo vector de controlo: modelo linear, modelo B spline e duas splines modelo Kochanek-Bartels (primeira spline: tensão = 0, continuidade = 0, bias = 0; segunda spline: tensão = -0.6, continuidade = 0, bias = 0)

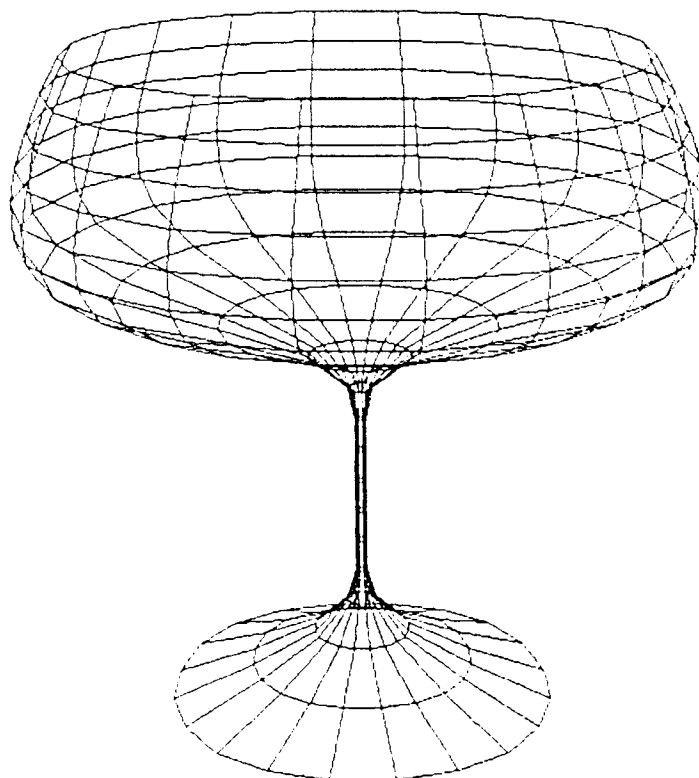


Fig. 3: Modelo criado com invocação de construtor geométrico propag: propagação de spline fechada ao longo de linha com escalamento controlado por spline.

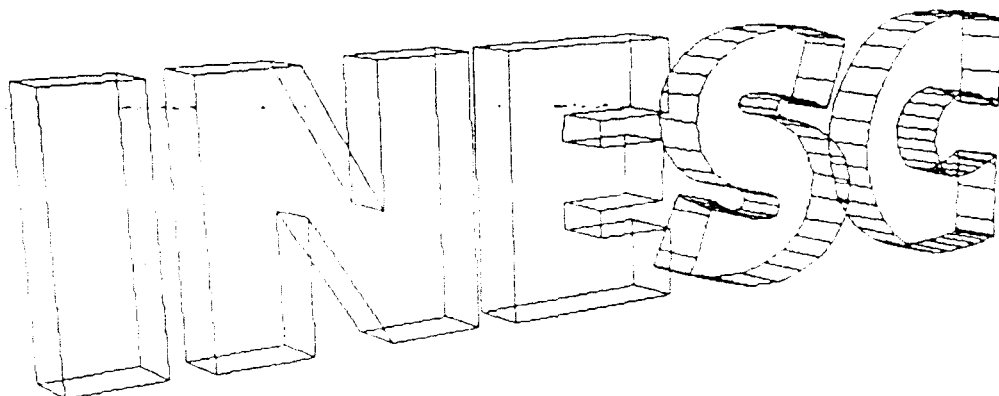


Fig 4: Exemplo de modelos criados através de extrusão utilizando o construtor geométrico propag.

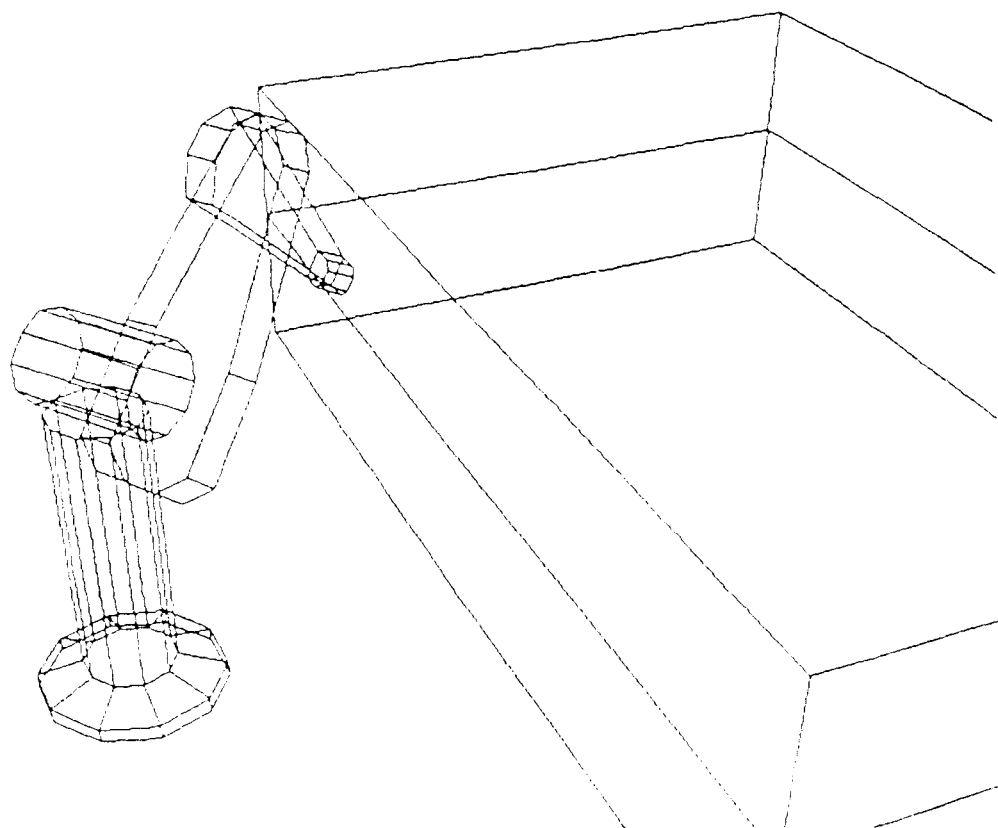


Fig 5: Braço de robot Puma com 4 graus de liberdade.

