

**Síntese de Imagens com
Elevado Nível de Realismo**

**António Augusto de Sousa, DEEC.FEUP/INESC.Norte
F. Nunes Ferreira, DEEC.FEUP/INESC.Norte**

**Primeiro Encontro Português de Computação Gráfica
LNEC, Julho de 1988
Lisboa**



Introdução

Na síntese de imagens destaca-se, pela elevada potencialidade de cálculo que exige, a obtenção de elevados níveis de realismo.

No topo de gama destes processos, encontram-se os algoritmos de "Ray-Tracing", conhecidos pelas imagens espectaculares que produzem. Apresentam, no entanto, o grave problema de consumirem tempos enormes de CPU, afastando a sua utilização das aplicações que se pretendem interactivas.

Apesar de dotados de um grau de realismo inferior, os algoritmos de visibilidade e de sombreamento podem produzir, em conjunto, imagens com uma qualidade aceitável e que se aproximam dos requisitos dos sistemas interactivos.

Esta comunicação pretende mostrar o trabalho realizado e em curso no INESC(Norte)/FEUP, na área da síntese de imagens com elevado nível de realismo, nomeadamente no cálculo de visibilidade de cenas 3D.

1.Trabalho Realizado

1.1-Objectivos e Requisitos

O trabalho realizado [Sou 87] compreende um algoritmo de cálculo de visibilidade em cenas 3D, podendo classificar-se como um algoritmo trabalhando no espaço imagem, à linha de varrimento ("Raster Scan Algorithm").

Os principais objectivos do trabalho foram a aquisição de "Know-How" na área do cálculo de visibilidade em cenas 3D e a definição de um algoritmo de visibilidade de fácil implementação em "Hardware". Assim, o algoritmo deveria ser simples e rápido, introduzindo o mínimo de restrições à cena a visualizar.

O algoritmo determina em cada linha de varrimento quais as porções visíveis das faces planas que compõem a cena, pelo que pode ser entendido com um algoritmo de **Remoção de Faces Ocultas** ("Hidden Surface Algorithm"). Desenvolveram-se ainda algumas ideias no sentido de, com o mesmo algoritmo, se conseguir efectuar a **Remoção de Linhas Ocultas** ("Hidden Line Algorithm").

1.2-Compreensão por Módulos do Algoritmo Implementado

O algoritmo começa com um **Bloco de Inicializações** no qual se inclui o maior número possível de operações. Posteriormente, entra num ciclo "linha a linha" no sentido de cima para baixo a que chamamos **Bloco de Cálculo de Visibilidade** e que se destina a obter os segmentos de faces visíveis em cada linha de varrimento, partindo de uma lista ordenada de arestas activas nessa linha (fig. 1).

Segue-se uma descrição breve dos três módulos que constituem o bloco de cálculo de visibilidade.

Módulo de Caracterização:

A determinação dos elementos (arestas e faces) numa linha de varrimento é reduzida à determinação das arestas activas. A estrutura de dados implementada permite, a partir de uma lista ordenada de arestas activas, determinar as faces activas.

Conforme se mostra na fig. 1, o módulo de caracterização divide-se em três sub-módulos: de Actualização, de Ordenação e de Inserção.

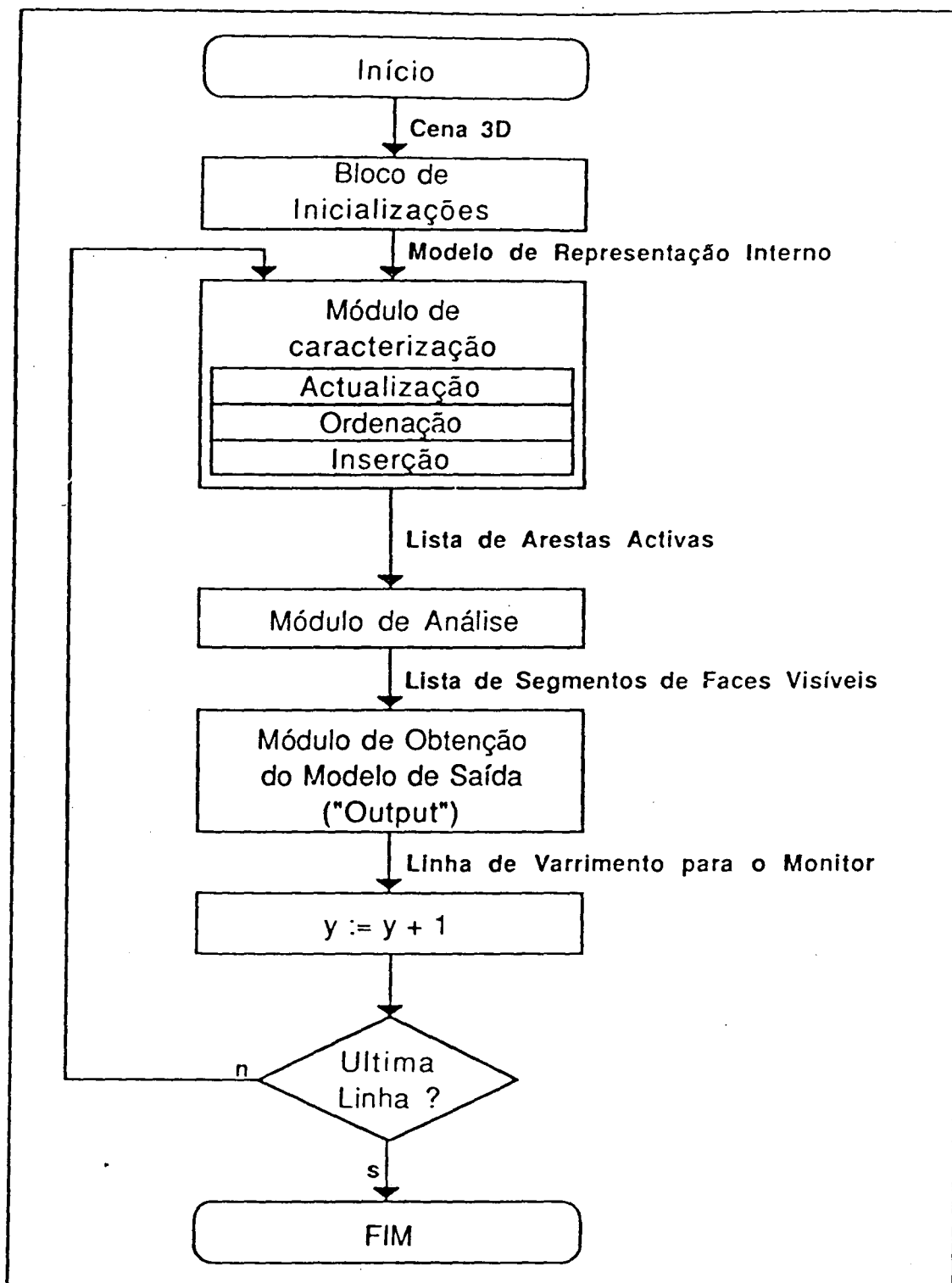


Fig. 1 - Módulos do Algoritmo de Visibilidade Implementado

O sub-módulo de **actualização** toma a lista de arestas activas que vem da linha de varrimento anterior e começa por eliminar arestas que deixam de ser activas (fig. 2, linhas 4, 7 e 9), após o que calcula as novas coordenadas de cada aresta sobrevivente.

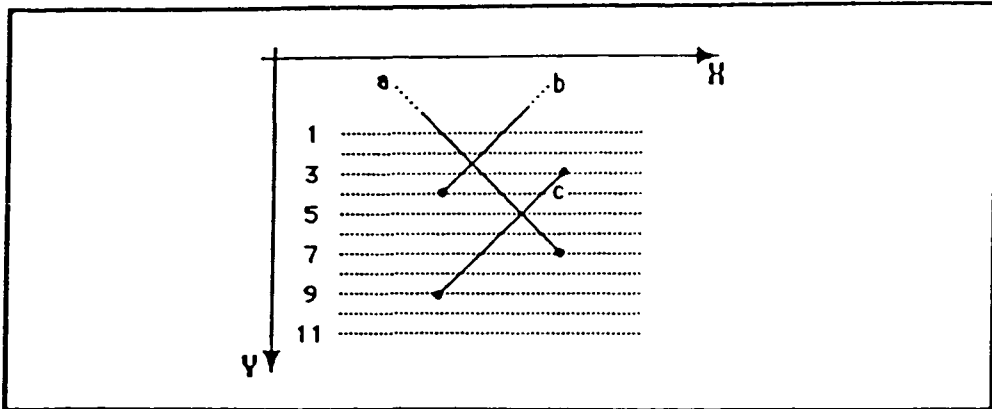


Fig. 2 - Projecção de três arestas

O sub-módulo de **ordenação** ordena a lista de arestas activas por ordem crescente da sua coordenada **X** (fig. 2, entre as linhas de varrimento 2 e 3, arestas **a** e **b**).

O sub-módulo de **inserção** é responsável pela introdução das novas arestas (fig. 1, linha 3, aresta **c**).

Módulo de Análise:

O módulo de análise recebe a **lista de arestas activas** já actualizada pelo módulo de caracterização e fornece uma **lista de segmentos de faces visíveis** nessa linha por ordem crescente de **X** (fig. 3).

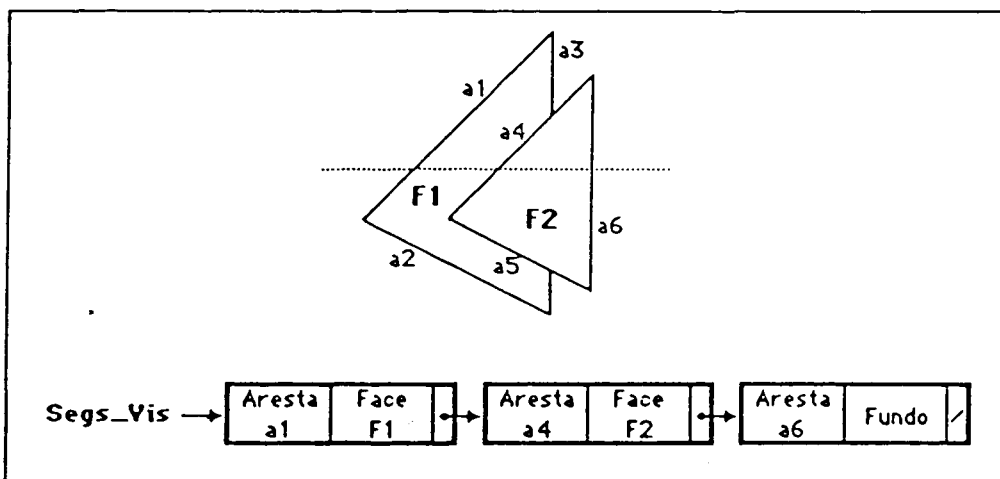


Fig. 3 - Obtenção da lista de segmentos visíveis Segs_Vis.

Na figura 4, mostra-se o plano **ZX** correspondente a uma linha de varrimento. Nesse plano, podem ver-se os segmentos de face correspondentes à intersecção do plano em questão com as faces **F1, F2, F3 e F4** e ainda as arestas que limitam essas faces **A1, A2, A7**.

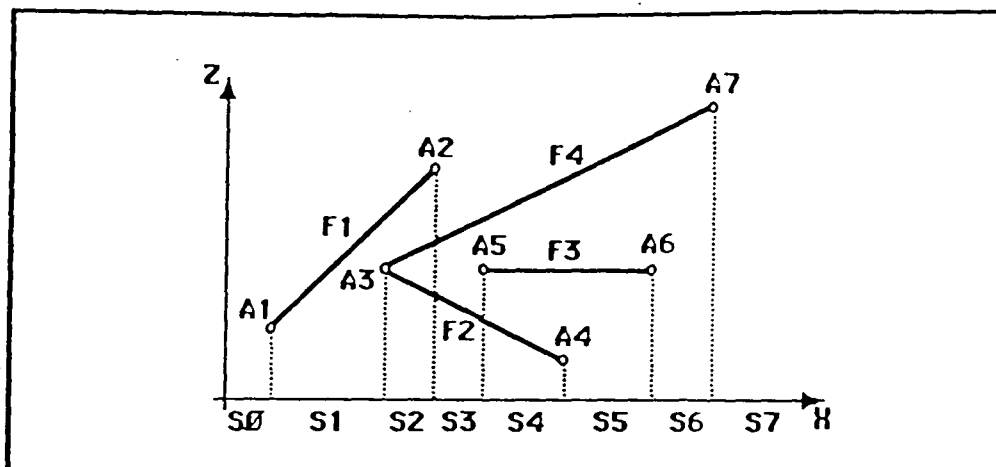


Fig. 4 - "Spans" numa linha de varrimento

Entre duas arestas consecutivas na lista de arestas activas, define-se um "Span" no interior do qual é necessário determinar quais as faces activas e decidir qual delas é visível. Para o efeito, o módulo de análise faz uso de uma fila de faces activas, ordenada por ordem crescente de profundidade, que é actualizada por cada cada aresta encontrada. Uma face pode passar a activa ou a não activa, por inversão do seu estado actual.

Módulo de Obtenção do Modelo de Saída:

Este módulo recebe como dados de entrada os elementos presentes na lista de segmentos de faces visíveis obtida no módulo de análise. Como resultado, fornece um modelo de descrição de uma linha de varrimento por meio de "coloração de segmentos de recta horizontais" (remoção de faces ocultas). Opcionalmente, pode ainda fornecer um "output" do tipo produção de desenho (remoção de linhas ocultas).

1.3-Aproveitamento da Noção de Coerência Vertical

No processo de varrimento de um monitor, encontram-se grandes grupos de linhas de varrimento cuja lista de segmentos visíveis é igual, sendo diferentes apenas as coordenadas X das arestas presentes na mesma. Para esse grupo de linhas diz-se que existe coerência vertical e efectua-se o módulo de análise uma só vez, na primeira linha do grupo. A quebra de coerência vertical pode acontecer em três situações detectáveis pelos três submódulos do módulo de caracterização: eliminação ou intersecção de

arestas já activas ou ainda o aparecimento de novas arestas que passam a ser activas.

O aproveitamento da coerência vertical permite que o módulo de análise seja efectuado somente nas linhas de varrimento onde exista quebra de coerência vertical.

1.4-Problemas não Resolvidos

O método adoptado para transformar o algoritmo de remoção de faces ocultas em remoção de arestas ocultas foi adaptado de um método proposto em [Bou 80] e funciona razoavelmente bem em grande parte das situações; algumas imprecisões nos cálculos conduzem a pequenos erros de desenho que o autor menciona e que tenta solucionar efectuando o estudo de visibilidade num domínio de maior resolução do que o necessário (por exemplo 1024 x 1024 em vez de 512 x 512 do ecran).

Porém, o método falha no desenho de arestas com uma inclinação muito pequena. Estas arestas podem, entre duas linhas de varrimento consecutivas, sofrer várias transições visível/invisível, enquanto que o algoritmo é capaz de detectar uma só transição. Como consequência, pode suceder que uma aresta apareça desenhada em zonas em que é invisível ou vice-versa e, no caso limite, pode suceder que uma aresta parcialmente visível nunca seja detectada como tal (fig. 5).

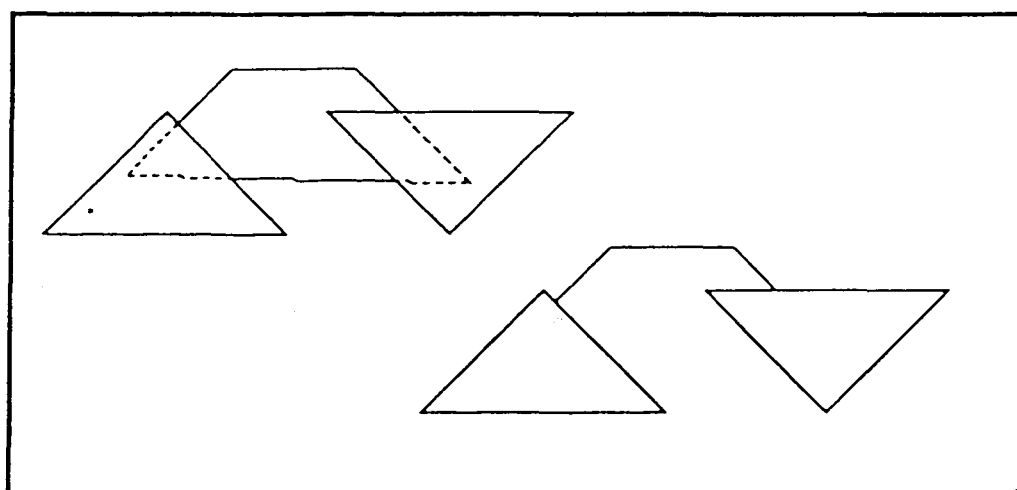


Fig. 5 - Desaparecimento de uma aresta parcialmente visível

Outro problema relaciona-se com a estrutura de dados utilizada. A simplicidade exigida por uma implementação "Hardware" levou-nos a adoptar uma estrutura de dados que acabou por se mostrar pouco flexível e com graves limitações de memória face à máquina utilizada no seu desenvolvimento.

2. Trabalho em Curso

2.1-Objectivos

Com os conhecimentos de cálculo de visibilidade adquiridos na versão anterior, tornava-se necessário desenvolver metodologias que resolvessem os problemas que subsistiam:

- desenvolvimento de uma nova estrutura de dados, dinâmica e, com a flexibilidade suficiente para permitir avaliar novas ideias
- aproveitamento dessa estrutura de dados para para obtenção de mais informação para além do cálculo de visibilidade
- desenvolvimento dos dois algoritmos, remoção de faces ocultas e remoção de linhas ocultas, sobre essa estrutura de dados

2.2-Novas Ideias em Desenvolvimento

Os problemas da versão anterior, relacionados com a produção de desenho, devem-se à existência de intersecções não detectadas entre duas linhas de varrimento (espaço imagem, valores de Y inteiros).

Em [SeW 85] refere-se um método proposto por Hammilton & Gear, capaz de eliminar esse tipo de problemas por não trabalhar ao nível de uma linha de varrimento mas sim ao nível de um vértice ou de uma intersecção. O método define um evento como podendo ser um **vértice**, uma **intersecção** ou uma **linha de varrimento**.

No processamento dos dois primeiros (vértice e intersecção), actualiza-se uma lista de arestas activas, partindo do pressuposto que somente a zona abrangida pelo evento é alterada.

O cálculo de intersecções aplica-se aos pares de arestas que passam a ser vizinhas após o tratamento de um vértice ou de outra intersecção (um par de

arestas de cada lado da zona afectada pelo evento). Não é portanto necessário procurar eventuais intersecções entre todos os pares possíveis de arestas. As intersecções calculadas são mantidas ordenadas por ordem crescente de Y (chave principal) e de X (chave secundária).

O evento Linha de Varrimento só é útil para o algoritmo de faces ocultas, podendo ignorar-se no algoritmo de linhas ocultas. Indica que o algoritmo se encontra sobre uma "Scan Line", pelo que é necessário enviar um descritor de linha para o monitor.

Algumas particularidades da nossa implementação:

1º Não existe o evento linha de varrimento. O algoritmo de faces ocultas funciona sobre uma estrutura pré-calculada.

2º A uma linha que contenha um ou mais vértices e/ou uma ou mais intersecções chamamos **Linha Interessante**. Pelo princípio de coerência vertical de uma cena 3D [Bou 80], só é necessário alterar as listas de elementos activos nas linhas interessantes.

3º A alteração das listas de elementos activos efectua-se em zonas restritas: porções afectadas por eventos.

4º Entre duas linhas de varrimento consecutivas, todos os eventos são detectados e as intersecções são calculadas com precisão.

5º Os descritores de linhas interessantes são mantidos em memória, numa estrutura de dados que, depois de completa, permite fácil e rapidamente obter uma imagem por porções de faces visíveis e/ou por porções de arestas visíveis.

Os pontos 2º e 3º permitem acelerar o processo de cálculo de visibilidade, enquanto que o ponto 4º elimina os problemas da versão anterior relacionados com as arestas de pequena inclinação. O ponto 5º é bastante exigente em termos de memória utilizada, mas permitirá obter informação adicional para além da informação de visibilidade.

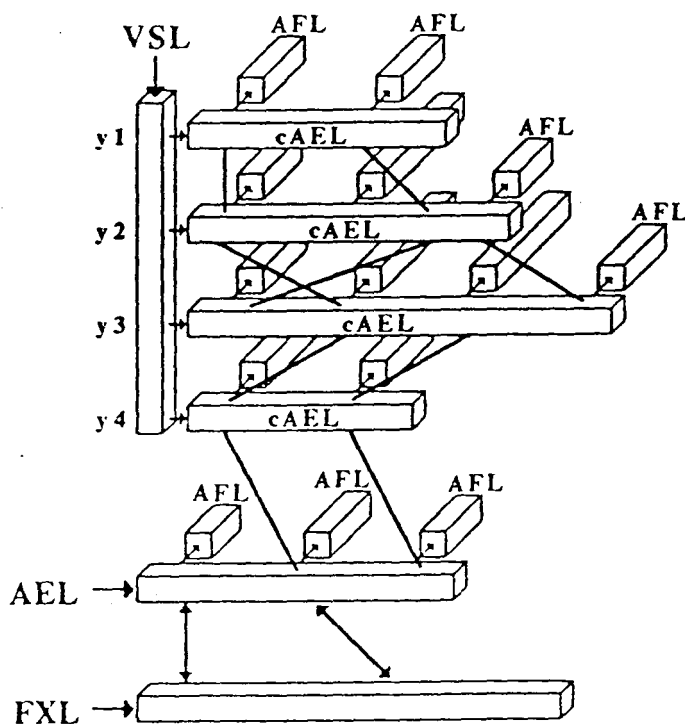
2.3-Estrutura de Dados

A estrutura de dados implementada é a representada na fig. 6. Cada prisma corresponde a uma lista ligada.



Os métodos utilizados para a obtenção desta estrutura são deduzidos de versões anteriores [Sou 87], complementadas com as novas ideias referidas na secção anterior.

Uma vez construída a estrutura VSL, é imediata a obtenção do modelo de visibilidade por segmentos de arestas visíveis ou por segmentos de faces visíveis, bastando para tal seguir os apontadores necessários. Na obtenção do modelo por segmentos de faces visíveis, a estrutura VSL é percorrida de cima para baixo, acedendo-se então às listas cAEL que são descritores das Linhas de Varrimento Interessantes. Com a informação contida nas listas cAEL, obtém-se um novo descritor que contém informação das faces visíveis nessa linha. Este descritor é válido até à próxima linha interessante (coerência vertical), a menos dos valores das coordenadas X dos contornos das porções visíveis de faces que deverão ser actualizados linha a linha.



- VSL Lista de apontadores por Linhas Interessantes que dão acesso a toda a estrutura
- AEL Lista de Arestas Activas ainda em actualização
- cAEL Listas de Arestas Activas já actualizada
- AFL Listas de Faces Activas por aresta
- FXL Lista de Intersecções Previstas

Fig. 6 - Estrutura de Dados em Desenvolvimento

Como exemplos de informação adicional a extrair da estrutura, refiram-se a obtenção dos contornos de corte por um plano horizontal e a realização de "Zooms". A primeira é facilitada pelas listas de faces activas AFL, enquanto que a segunda se reduz praticamente ao cálculo de um factor de escala.

A estrutura de dados e os algoritmos necessários estão a ser desenvolvidos por dois alunos finalistas do curso de Engenharia Electrotécnica e de Computadores da FEUP: Clemente Luis Sequeira Aguiar e José Alberto Quintal de Freitas.

3. Perspectivas Futuras

A investigação na área de visualização de cenas 3D continua a ser uma realidade, mesmo em centros de investigação de renome mundial. O grupo de CG & CAD do INESC(Norte) tenciona continuar a investir neste campo, nomeadamente nos algoritmos de visualização de imagens realistas baseados em métodos de "Ray-Tracing" [DiS 84] e de "Radiosity" [GTG 84], [CGr 85], [ICG 85].

O trabalho realizado no âmbito do cálculo de visibilidade continuará, justificando-se o seu desenvolvimento pela maior rapidez de resposta deste tipo de algoritmos. Assim, o algoritmo de visibilidade baseado na estrutura descrita na secção anterior será integrado no modelador de sólidos MS(E) [Bra 88] e ser-lhe-ão adicionados outros módulos cuja natureza dependerá do tipo de informação extra que se conseguir extrair da estrutura VSL. Está também previsto incluir um módulo de Iluminação Suavizada ("Smooth Shading") que permita a representação de superfícies curvas por faces planas [BiW 86].

O aparecimento de novas tecnologias de processamento paralelo, nomeadamente a tecnologia de implementação com "Transputers" [INM 87], permite-nos pensar na adaptação dos métodos conhecidos e/ou no desenvolvimento de novos métodos de "Ray-Tracing" e de "Radiosity", com menores tempos de cálculo.

4. Bibliografia

[BiW 86] Gary Bishop, David M. Weimer

Fast Phong Shading

ACM Computer Graphics, Vol 20, Nº 4, Julho 1986

[Bou 80] Ph. Boule

Etude et Realisation d'Algorithmes pour la Visualisation de Scènes
Composées de Facettes Planes

Thèse de Docteur-Ingénieur, Grenoble, 1980

[BrF 88] Vasco Branco, F. Nunes Ferreira

Um Modelador de Sólidos Experimental: o MS(E)

1º Encontro Português de Computação Gráfica

Lisboa, Julho de 1988

[CGr 85] Michael Cohen, Donald Greenberg

The Hemi-Cube: A Radiosity Solution for Complex Environments

ACM Computer Graphics, Vol 19, Nº 3, Julho 1985

[DiS 84] Mrk Dippé, John Swensen

An Adaptative Subdivision Algorithm and Parallel Architecture
for Realistic Image Syntesis

ACM Computer Graphics, Vol 18, Nº 3, Julho 1984

[GTG 84] Cindy Goral, Kenneth Torrance, D. Greenberg, Bennett Battaile

Modeling Interaction of Light Between Diffuse Surfaces

ACM Computer Graphics, Vol 18, Nº 3, Julho 1984

[ICG 85] David Immel, Michael Cohen, Donald Greenberg

A Radiosity Method for Non-Difuse Environments

ACM Computer Graphics, Vol 20, Nº 4, Julho 1986

[INM 87] The Transputer Family

INMOS Product Information, 1987

[SeW 85] Carlo H. Sequim, Paul R. Wensley
Visible Feature Return at Object Resolution
IEEE CG&A, Vol 5, Nº 5, Maio 1985

[Sou 87] António Augusto de Sousa
Cálculo de Visibilidade em Cenas 3D: Síntese de Conceitos, Implementação de
um algoritmo
Provas de Aptidão Pedagógica e Capacidade Científica-FEUP, Porto, 1987

[WCG 87] John Wallace, Michael Cohen, Donald Greenberg
A Two-Pass Solution to the Rendering Equation:
A Synthesis of Ray Tracing and Radiosity Methods
ACM Computer Graphics, Vol 21, Nº 4, Julho 1987