

FAROP: Um Sistema de Apoio ao Ensino da Visualização Tridimensional

João A. Madeiras Pereira

IST/INESC

jap@inesc.pt

SUMÁRIO

Este artigo descreve a arquitectura do sistema FAROP (Ferramenta de Apoio à Rasterização de Objectos Poligonais) o qual constitui uma plataforma genérica para o ensaio de algoritmos de rasterização de objectos poligonais implementados por alunos nos Laboratórios de diversas cadeiras na área de Computação Gráfica no Instituto Superior Técnico. Este sistema permite ainda não só alimentar algoritmos de rasterização mas também modelar cenas tridimensionais em formato NFF [Haines87] e SIPP [Yngvesson92] e realizar a síntese das respectivas imagens.

1. Objectivos

O *pipeline* de visualização 3D constitui um tópico de ensino fundamental nas cadeiras na área de Computação Gráfica leccionadas no Instituto Superior Técnico, nomeadamente a disciplina de opção Computação e Métodos Gráficos do 5º ano do Curso de Engenharia de Electrotécnica e Computadores, a disciplina Computação Gráfica do 3º ano do Curso de Engenharia de Informática e Computadores e a disciplina Introdução à Computação Gráfica do Mestrado em Engenharia de Electrotécnica e Computadores. Nesta perspectiva, o ensino teórico deste tópico nas referidas cadeiras é reforçado com um trabalho laboratorial cujo conteúdo incide no último andar do *pipeline* de visualização, nomeadamente a rasterização onde os alunos têm de implementar dois algoritmos para remoção de superfícies ocultas, z-buffer e algoritmo de *Watkins*, e o algoritmo *Gouraud* para a execução do sombreamento.

Impunha-se, deste modo, dispor de uma plataforma genérica que gerasse ficheiros contendo descrições de cenas tridimensionais em Coordenadas de Ecrã para o ensaio intensivo de algoritmos de rasterização de objectos poligonais. Assim, foi desenvolvido, no âmbito de um Projecto Final de Curso em Engenharia Electrotécnica e Computadores, um sistema que satisfaz

os requisitos pretendidos. O sistema FAROP [Baptista95, Pereira96] permite definir e desenvolver os componentes necessários ao ensaio de um sub-sistema de rasterização, recorrendo a descrições de cenas tridimensionais disponíveis sob determinados formatos. Em particular, procurava-se recorrer a descrições de cenas disponíveis em ficheiros de dados com o formato NFF (*Neutral Format File*) para alimentar e testar algoritmos de rasterização desenvolvidos quer no contexto de aprendizagem de técnicas clássicas quer no contexto de investigação de novas estratégias de rasterização. Adicionalmente, com o fim de cumprir o objectivo de alargar a potencialidade e a flexibilidade do ambiente de teste, optou-se por dotar o FAROP de capacidades que lhe valessem processar cenas descritas a um nível mais alto, sob a forma de uma outra linguagem de descrição de cenas tridimensionais, denominada de SIPP (*Simple Polygon Processor*). A escolha dos formatos NFF e SIPP justifica-se pelo facto de estes estarem associados à necessidade de encontrar um referencial comum para a análise do desempenho de diferentes arquitecturas de *hardware* ou diferentes algoritmos (por exemplo, *ray-tracing*, *radiosidade*, *pipeline* de visualização). Na realidade, um dos primeiros esforços de normalização desenvolvidos nesta área, que se tornou familiar no universo da Computação Gráfica, consistiu na criação de um ambiente de testes denominado *Standard Procedural Database (SPD)* [Haines87]. Trata-se de um conjunto de programas que permitem a geração de ficheiros de dados contendo descrições de cenas tridimensionais num formato simples designado por *Neutral Format File*. O NFF permite a especificação de parâmetros de visualização, de características e posicionamento de fontes luminosas, de primitivas gráficas elementares (polígonos, esferas, cones e cilindros) e de características físicas das superfícies (côr, índices de reflexão da luz, etc.) dos objectos (ver Apêndice A.1). Além disso, foi disponibilizado para o domínio público uma biblioteca gráfica que constitui um ambiente bastante versátil para a modelação e visualização de objectos complexos, o SIPP [Yngvesson92]. Esta biblioteca inclui funções específicas para a modelação hierárquica de cenas: hierarquização de objectos, definição de primitivas gráficas de alto nível (cubo, prisma, esfera, elipsóide,

cilindro, cone, *torus*, etc.) e a especificação de parâmetros de visualização, de fontes luminosas e características de superfícies.

A arquitectura do sistema de teste foi concebida de modo a ser suficientemente flexível, permitindo assim, de forma totalmente homogénea e transparente, a manipulação e integração de objectos descritos nas duas linguagens e a sua visualização de forma simultânea (*Scene Merge*).

Para tal, foram desenhados e desenvolvidos os componentes de *software* que permitissem:

- ler e interpretar a descrição da cena (objectos, características da iluminação, ponto de observação, etc.) especificada em NFF e/ou SIPP;
- gerar uma base de dados de objectos poligonais;
- implementar as operações de transformação geométrica;
- produzir as *interfaces* necessárias ao acoplamento do motor de rasterização externo (chamada à rotina de rasterização fornecida pelo utilizador ou criando um ficheiro de saída em Coordenadas de Ecrã).

2. Metodologia de ensaio

É importante clarificar a ideia chave subjacente a esta noção de "acoplamento ao motor de rasterização externo" já que ela pretende abarcar as principais metodologias de ensaio a esquemas de rasterização.

Assim, um projectista poderá desejar desenvolver e, eventualmente, avaliar diferentes abordagens ao problema da rasterização, sem ter de se preocupar com a plataforma de execução. Neste caso particular, o procedimento de teste, utilizando o FAROP, consiste na inclusão da rotina que implementa o algoritmo de rasterização em causa para posterior invocação. Num outro modo de teste, em que a plataforma de execução é relevante (e, muito provavelmente, diferente daquela em que o sistema de teste se encontra instalado), a metodologia mais simples consiste na geração de um ficheiro de saída contendo a descrição da cena em Coordenadas de Ecrã (ver exemplo no Apêndice A.2), o qual será posteriormente utilizado na máquina na qual se encontra o algoritmo de rasterização a ser ensaiado. **Esta última metodologia de teste configura o quadro experimental em que o sistema foi, até agora, utilizado nos laboratórios das disciplinas de Computação Gráfica: ensaio dos algoritmos desenvolvidos pelos alunos com os ficheiros de teste produzidos pelo FAROP em Coordenadas de Ecrã.**

O FAROP possibilita, ainda, a capacidade de visualização das cenas, ao integrar, por defeito, um sub-sistema de rasterização baseado nos algoritmos do

programa SIPP. A sua invocação é uma opção de programa disponibilizada ao utilizador e que se revela particularmente útil em situações cujo objectivo é, unicamente, visualizar uma imagem decorrente da especificação (modelação) de uma cena.

No futuro, na posse das necessárias infraestruturas, pretende-se disponibilizar o sistema completo aos alunos de modo a que estes tenham acesso a todas as potencialidades do FAROP para modelação e visualização tridimensional.

3. Arquitectura global

O diagrama de blocos que define a arquitectura global do sistema FAROP está ilustrado na figura 1.

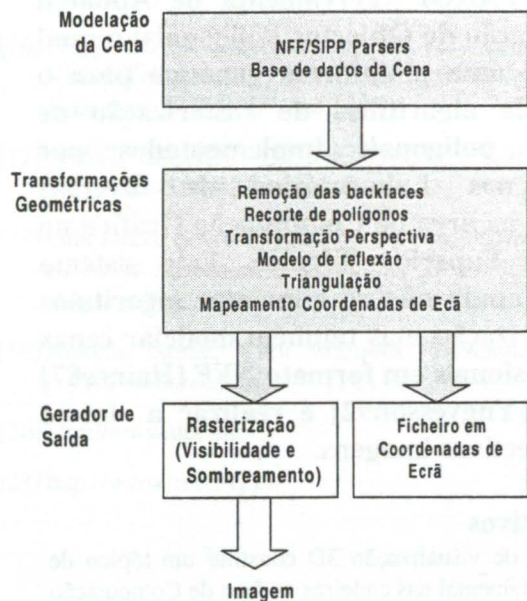


Figura 1 - Diagrama de blocos do sistema FAROP

O sistema de teste é constituído por três grandes blocos que serão discutidos detalhadamente na próxima secção:

- as operações de construção da cena, que integram a interpretação (*parsing*) das linguagens NFF e SIPP e o carregamento da base de dados de objectos e parâmetros de visualização;
- as operações geométricas que actuam, basicamente, sobre os vértices dos polígonos e que incluem as operações de pré-remoção de polígonos não visíveis (*Culling* ou *Backface Polygon Removal*), o recorte de polígonos (*Clipping*), a aplicação da transformação perspectiva, a determinação da luminosidade por aplicação do modelo de iluminação, a segmentação opcional (*Polygon Splitting*) de polígonos em triângulos (ou triangulação), e a operação de mapeamento nas coordenadas físicas (inteiras) do dispositivo de visualização, (*Device Dependent Coordinates*);
- as operações geradoras de informação final que incluem, por opção do utilizador, a construção de um ficheiro com a informação dos polígonos da cena em

Coordenadas de Ecrã, destinado a alimentar um sistema de rasterização totalmente externo ao ambiente de teste, e/ou a rasterização (algoritmo externo fornecido pelo utilizador ou o algoritmo interno disponibilizado, por defeito, pelo sistema) responsável pelo cálculo dos pixels, remoção de superfícies ocultas (*Hidden Surface Removal*), e sombreado (*Shading*). Neste último caso, o ambiente de teste comporta-se como um *pipeline* de visualização completo.

O sistema FAROP faculta, ainda, algumas características adicionais que podem ser essenciais na sua função de plataforma genérica para suporte de visualização de polígonos. Nomeadamente:

- aos métodos de sombreado *Phong* e *Gouraud* junta-se a possibilidade de visualizar os objectos com sombreado constante ao longo do polígono (*Flat Shading*) ou simplesmente em modelo de "arame" (*Wire-Frame* ou *Line*) utilizando o algoritmo interno de rasterização (ver Apêndice B.1);
- as estruturas internas de dados podem ser exportadas sob um formato NFF, (*NFF File Output*) permitindo, por exemplo, a conversão de cenas descritas em linguagem SIPP em ficheiros de dados de formato NFF;
- o sistema permite a segmentação de polígonos com um número variável de vértices em triângulos (triangulação). Esta facilidade permite alimentar motores de rasterização otimizados (ou limitados) ao processamento deste tipo de primitivas geométricas;
- por fim, o sistema, oferece um conjunto de ferramentas que permitem contabilizar os consumos, em termos de recursos da máquina, dos principais andares do *pipeline*. Esta facilidade permite uma primeira avaliação do desempenho dos algoritmos e/ou máquinas (*System Tuning*).

Resta apenas referir que, apesar do sistema FAROP se encontrar presentemente instalado numa máquina UNIX, ele é facilmente portátil para qualquer tipo de computador sequencial que possua um compilador de C.

4. Arquitectura detalhada

Em seguida, proceder-se-á a uma descrição mais detalhada acerca dos aspectos mais relevantes da arquitectura interna do sistema FAROP, bem como da sua funcionalidade.

4.1. Conceitos básicos

Apesar do FAROP facultar a um utilizador a capacidade de descrição de cenas de complexidade elevada através do recurso a primitivas gráficas de alto nível que os formatos NFF e SIPP disponibilizam, estas são "pré-processadas" pelo sistema, de forma a serem modeladas em termos de polígonos. O sistema permite exclusivamente o processamento de objectos descritos sob a forma de conjuntos de polígonos planares,

convexos ou côncavos e sem buracos, processamento esse que implica, em primeiro lugar, a aplicação de um conjunto de transformações geométricas às quais se encontram associados os respectivos sistemas de coordenadas.

De forma a facilitar a modelação de um determinado objecto, opta-se geralmente por descrever os seus diversos componentes (isto é, especificar os vértices dos polígonos que o constituem) em função de um ponto localizado dentro ou perto do próprio objecto. Nesta situação, os objectos dizem-se descritos em termos de um **Sistema de Coordenadas Local** (ou *Local Coordinate System*).

Após a modelação, o próximo passo é colocar o objecto na **cena** que se pretende visualizar. Cada objecto que constitui a cena tem o seu referencial local. Todos os objectos são transformados de forma a poderem ser representados em função de um referencial único, o que permite especificar o seu posicionamento relativo. Este referencial comum é denominado **Sistema de Coordenadas do Mundo** (ou *World Coordinate System*). Ainda neste sistema de coordenadas, a descrição da cena é complementada com a especificação das posições das fontes luminosas e da posição da câmara, em particular, do ponto de observação e da direcção de observação.

Em Computação Gráfica, normalmente, recorre-se à noção conceptual de **câmara virtual** [Watt93] para definir um novo referencial, denominado **Sistema de Coordenadas da Câmara** (*Eye or Camera Coordinate System*), cuja origem coincide com o centro de projecção (posição da câmara) e cujo eixo dos zz' é normal ao plano de visualização (*View Plane*) e representa a direcção de observação.

Por fim, o último espaço de coordenadas em que os objectos são transformados (talvez o menos intuitivo) está relacionado com as técnicas de projecção dos objectos da cena no plano de visualização. Define-se, nesse sentido, um **volume de visualização** (*Viewing Frustum*) que delimita o volume do espaço que será visualizado. A referida transformação com carácter projectivo é denominada de **Transformação Perspectiva** e os objectos resultantes são expressos em termos de Coordenadas de Ecrã (*Screen Coordinates*). Em rigor, as coordenadas resultantes desta transformação são normalizadas pelo que se impõe um mapeamento destas no referencial determinado pelo dispositivo físico de visualização (*Device Dependent Coordinates*).

4.2. Estrutura de dados da cena

Uma cena é composta por objectos, parâmetros de visualização e fontes luminosas. Um determinado objecto complexo pode ser modelado recorrendo a outros objectos de maior simplicidade, ou seja, um objecto pode referenciar sub-objectos. A forma natural

de armazenar a informação respeitante aos objectos que compõem uma determinada cena é sob a forma de uma estrutura de dados em árvore, em que cada nó representa um objecto.

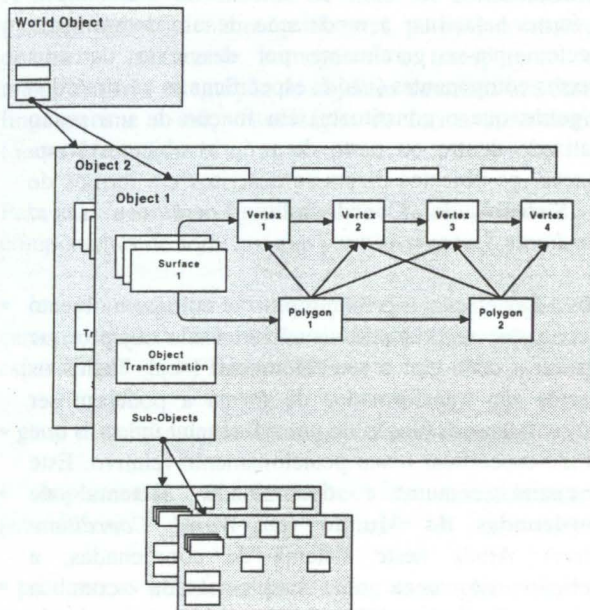


Figura 2 - Estrutura hierárquica dos objectos na cena

Cada objecto é por sua vez definido por uma lista de superfícies que, para além das suas características físicas (côr, opacidade, parâmetros de iluminação, etc.), contém uma lista de polígonos e uma lista de vértices. A estrutura de dados é otimizada de modo a permitir a partilha de vértices de polígonos de uma mesma superfície. Deste facto, advêm economias óbvias, no que respeita ao volume de informação a armazenar.

Com o objectivo de facilitar as tarefas de modelação da cena, os objectos são definidos em termos de coordenadas locais, aos quais se associam transformações geométricas (translações, rotações, etc.) que permitem definir o seu posicionamento relativamente ao referencial universal. Estas transformações, representadas internamente em termos de matrizes, são conjugadas e armazenadas como parte integrante da estrutura de dados do objecto, sendo aplicadas posteriormente aquando do processamento geométrico do objecto¹. Cada objecto mantém na sua estrutura de dados uma matriz de transformação relativamente ao seu objecto superior em termos de hierarquia. Qualquer transformação geométrica aplicada a um objecto propaga-se de forma recursiva a todos os seus sub-objectos. Refira-se que os polígonos são caracterizados por possuírem duas faces, a frontal e a traseira (*Front Face* e *Back Face*). A ordem pela qual

se definem os respectivos vértices permite identificar as suas faces.

A figura 2 representa, de forma esquemática, a estrutura interna de dados utilizada no armazenamento da descrição dos objectos que compõem a cena. Neste modelo, cada polígono é descrito internamente com base em:

- uma lista de estruturas em que cada uma contém a informação de um vértice, em coordenadas locais, relativa à sua posição espacial e à sua normal;
- a normal ao plano que contém o polígono, definido também no espaço de coordenadas locais. Este vector é utilizado, nomeadamente, durante a operação de remoção de faces traseiras.

4.3. Operações elementares

A figura 3 apresenta de forma esquemática as diversas operações elementares que implementam a funcionalidade providenciada pelo sistema FAROP. As operações encontram-se agrupadas em três grandes blocos: **Modelação e construção da base de dados da cena**, **Transformações geométricas** e **Gerador de saída**. Para fácil interpretação, cada operação encontra-se identificada por um número e é descrita em detalhe nas sub-secções seguintes.

4.3.1. Modelação e construção da base de dados da cena

O bloco de modelação e carregamento das estruturas internas de dados (operações 1, 2 e 3) engloba as funções de interpretação dos ficheiros de dados NFF e o processamento das definições SIPP.

O primeiro conjunto de operações executa o carregamento da base de dados da cena com a informação proveniente de um ficheiro NFF: as características das superfícies, os polígonos² que constituem essas superfícies, os parâmetros de visualização, a definição de fontes de luz, as dimensões (ou resolução) da imagem final, etc. De realçar que, a totalidade das superfícies é armazenada internamente sob a forma de um único objecto o qual poderá ser posteriormente manipulado por aplicação de transformações geométricas.

O segundo conjunto tipo de operações engloba o carregamento dos objectos SIPP na estrutura de dados internos, bem como as transformações geométricas que lhes estão associadas. Note-se que, de igual forma ao que se passa com as definições NFF, a linguagem SIPP permite, para além da definição de objectos, a especificação de parâmetros de visualização, fontes luminosas e características de superfícies.

¹No caso de uma descrição em formato NFF, todos os objectos são representados em coordenadas do Mundo Real.

²As primitivas esfera, cone ou cilindro são pré-processadas e decompostas em polígonos.

Todas as especificações de objectos, fontes de luz, etc., provenientes quer do ficheiro de definições NFF quer da linguagem SIPP se complementam, com excepção da definição dos parâmetros de visualização (posição da

câmara, cor de "background") que se sobrepõem, pelo que, a ordem pela qual as definições surgem no programa principal tem obviamente impacto no valor final desses parâmetros.

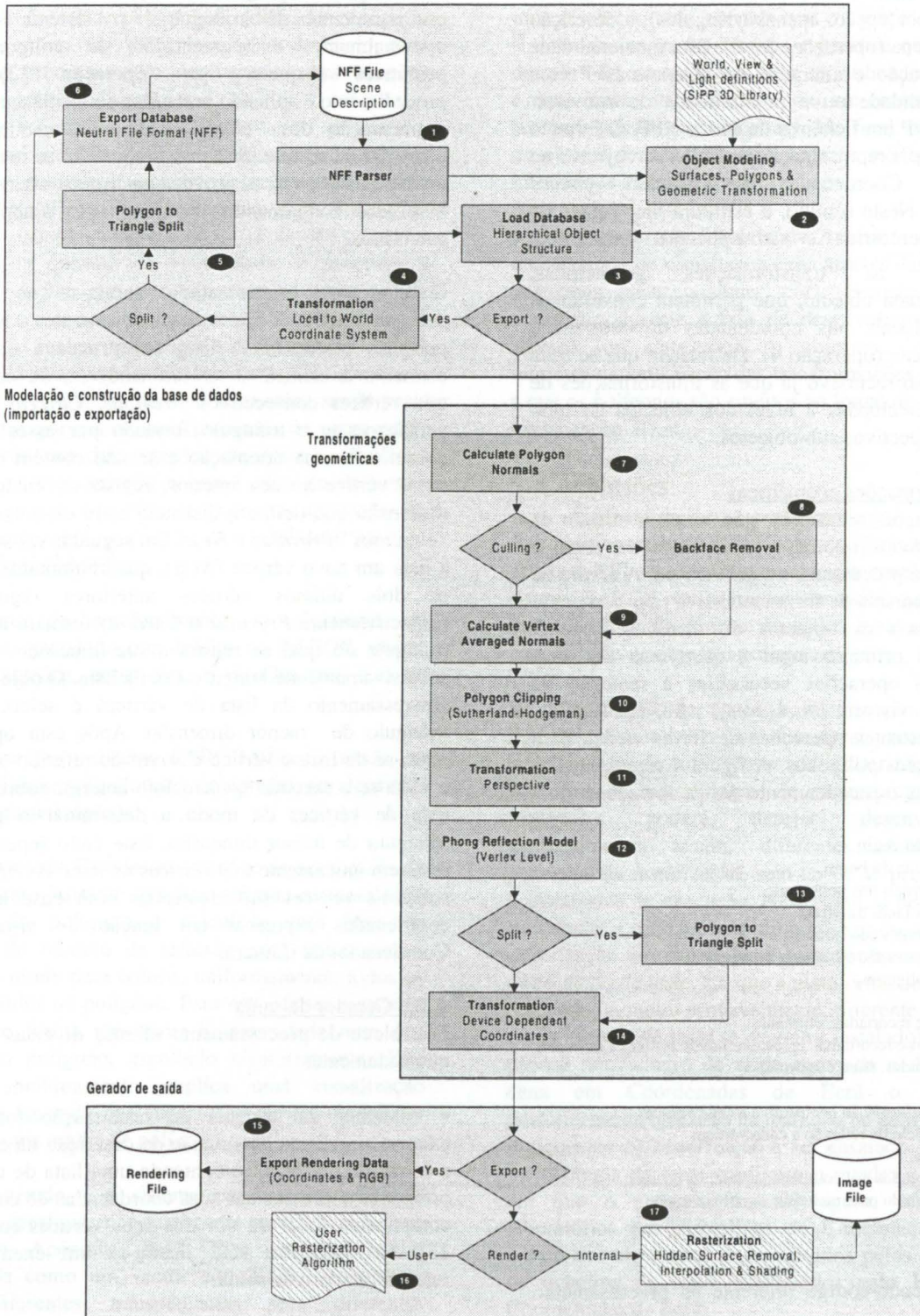


Figura 3 - Fluxo de actividades do sistema FAROP

No final da fase de modelação e carregamento, todas os objectos quer importados do ficheiro de entrada NFF, quer definidos através da linguagem SIPP, encontram-se

armazenadas na estrutura em árvore já descrita anteriormente (ver figura 2) e podem ser processadas,

de uma forma totalmente transparente, pelos blocos subsequentes do FAROP.

O sistema tem a possibilidade de exportar, sob a forma de um ficheiro NFF (contendo exclusivamente polígonos e, por opção, apenas triângulos), a descrição completa da cena (operações 5 e 6). Esta funcionalidade permite a geração e alteração de ficheiros NFF com grande flexibilidade ou a possibilidade de converter programas SIPP em ficheiros de dados NFF. O formato NFF pressupõe a representação de todos os objectos no espaço de Coordenadas de Mundo (*World Coordinates*). Neste sentido, a estrutura hierárquica de objectos é percorrida e são aplicadas (se houver necessidade)³ as transformações geométricas, associadas a cada objecto, que permitem converter as coordenadas locais nas coordenadas do referencial universal da cena (operação 4). De realçar, que se trata de um processo recursivo já que as transformações de coordenadas efectuadas a nível dos objectos de topo afectam os respectivos sub-objectos.

4.3.2. Transformações geométricas

Após as operações anteriores, que visam a criação da estrutura de dados interna representativa da cena, o sistema inicia o processamento geométrico. A árvore de objectos é percorrida de forma recursiva (*top-down*) e o sistema processa os objectos de forma sequencial, aplicando, em primeiro lugar a cada uma das suas superfícies, as operações necessárias à remoção de polígonos não visíveis (*backfaces*), para, em seguida, executar as restantes operações de forma cíclica para cada um dos seus polígonos. A figura 4 representa, de forma resumida, o encadeamento destas operações.

```

for each objecto do begin
  for each superfície do begin
    calcular as normais nos polígonos;
    eliminação das faces traseiras;
    calcular as normais aos vértices;
  for each polígono não eliminado do begin
    recorte de polígonos;
    case begin
      totalmente recortados: eliminar;
      parcialmente recortados: recalcular novos vértices e normais;
      não recortados: then continue;
    end;
    aplicar a transformação perspectiva a cada vértice;
    aplicar o modelo de reflexão a cada vértice;
    triangulação;
    vértices mapeados nas coordenadas inteiras do ecrã;
    polígono enviado para rasterizar ou armazenado;
  end;
end;
end;

```

Figura 4 - Pseudo-código referente ao processamento geométrico

³Apenas, aos objectos previamente definidos com base em primitivas SIPP.

Os polígonos são submetidos ao algoritmo *Sutherland-Hodgman* para recorte tridimensional de polígonos côncavos ou convexos [Foley90] (ver Apêndice B.2).

Com o objectivo de alimentar algoritmos de rasterização otimizados (ou limitados) ao processamento de triângulos, o sistema permite, opcionalmente, a segmentação de polígonos em primitivas daquele tipo (operação 13). Este procedimento é aplicado nesta fase de forma a garantir a segmentação dos “novos” polígonos resultantes da operação de recorte. Sublinhe-se que não se introduzem novos vértices ou se provoca quaisquer alterações ao nível das componentes de iluminação anteriormente calculadas.

O algoritmo implementado baseia-se no método descrito em [Judd88] e utiliza, novamente, a noção de polígono orientado. A lista de vértices é percorrida elemento a elemento, seleccionando-se, de cada vez, três vértices consecutivos *Previous*, *Current* e *Next*: verifica-se se o triângulo formado por esses vértices possui a mesma orientação e se não contém qualquer outro vértice no seu interior; regista-se, então, a sua **dimensão** que define a distância entre os dois vértices “extremos” *Previous* e *Next*. Em seguida, vai-se buscar à lista um novo vértice (*Next*), que conjuntamente com os dois últimos vértices anteriores (agora são respectivamente *Previous* e *Current*) formam um novo triângulo do qual se regista a sua dimensão, e assim sucessivamente até atingir o fim da lista. O objectivo do atravessamento da lista de vértices é seleccionar o triângulo de menor dimensão. Após esta operação, retira-se da lista o vértice *Current* do referido triângulo e volta-se a executar o raciocínio anterior sobre a nova lista de vértices de modo a determinar o próximo triângulo de menor dimensão. Este ciclo repete-se até restarem unicamente três vértices na lista. As operações sobre os vértices são efectuadas com base nas suas coordenadas expressas em função do sistema de Coordenadas da Câmara.

4.3.3. Gerador de saída

Este bloco de processamento oferece diversas opções, nomeadamente:

- alimentar um sistema de rasterização totalmente externo através da exportação da descrição da cena sob a forma de um ficheiro contendo uma lista de todos os polígonos expressos nas suas coordenadas de ecrã e nas componente RGB de cor dos seus vértices (operação 15). No Apêndice A.2, ilustra-se um exemplo do referido ficheiro de saída;
- não executar rasterização;
- utilizar o algoritmo interno de rasterização (operação 17) e que permite executar as operações de cálculo de pixels, remoção de faces ocultas e sombreadimento. O sistema produz ficheiros em formato **ppm** (*Portable*

Pixel Map) passíveis de serem visualizados por ferramentas de uso genérico e apresenta, no ecrã, informação de diagnóstico (ver Apêndice A.3)

- o acoplamento directo de um algoritmo externo de rasterização fornecido pelo próprio utilizador (operação 16). Essa rotina faz parte integrante do ciclo mais interno de processamento que opera numa base de polígono a polígono, ver figura 4.

O algoritmo interno de rasterização utiliza diversas formas de coerência espacial (coerência de aresta, coerência de linha de varrimento) de modo a se utilizarem cálculos incrementais [Foley90]. Para a resolução da questão de visibilidade, é proposto o esquema *z-buffer* que é muito simples ao nível da programação mas com elevados requisitos em termos de memória. De realçar que, segundo este método, alguns polígonos são desenhados no *Frame Buffer* (isto é, são aplicadas as operações de sombreamento) sendo mais tarde ocultos por outros mais próximos do observador. Este é, aliás, um dos aspectos de maior ineficiência associados a este método e que é alvo de propostas alternativas cuja filosofia consiste na aplicação do sombreamento após a questão da visibilidade ter sido solucionada [Tebbs92].

Em termos de sombreamento, o algoritmo de rasterização, instalado por defeito no FAROP, utiliza diversos métodos, entre eles, o de *Gouraud* e o de *Phong*.

Adicionalmente, o sistema FAROP oferece, ainda, ao nível da visualização (funções integradas no algoritmo interno de rasterização):

- sombreamento Constante** - trata-se de um método de sombreamento bastante simples, em que a intensidade luminosa do polígono é calculada com base na aplicação do modelo de reflexão num único ponto, sendo esta usada para colorir, uniformemente, todos os pixels contidos no polígono. Para este efeito, o sistema recorre (de forma grosseira) à intensidade num dos vértices do polígono, escolhido aleatoriamente. Este tipo de sombreamento implica uma visualização claramente facetada das superfícies curvas modeladas através de uma malha de polígonos;

- transparência** - o sistema permite especificar e rasterizar superfícies não totalmente opacas (ver a cena *Mount* no Apêndice B.1). Esta característica é armazenada como um vector *côr* (RGB), permitindo definir diferentes transparências para diferentes comprimentos de onda. Os valores desta grandeza "opacidade" podem variar entre 0 e 1, em que 1 representa uma superfície totalmente opaca e 0 corresponde a uma superfície totalmente transparente (invisível);

- mecanismo de anti-aliasing** - o sistema permite a aplicação do método de **Supersampling** para a redução de ruído. Em traços gerais, este método mantém a metodologia normal de rasterização inalterada e passa a calcular a solução final num espaço de ecrã virtual com uma resolução mais elevada que o espaço de ecrã real. É definido um parâmetro que especifica qual o grau **A** de sobre-amostragem que deve ser utilizado. Cada pixel é rasterizado internamente como um conjunto de ($A \times A$) "sub-pixels" (ou amostras). No final, a intensidade do pixel é determinada com base na média simples da intensidade das suas amostras;

- visualização em Modelo de Arame** - um método muito rápido de visualizar a cena através das arestas dos polígonos (sem qualquer *côr*). Permite facilmente visualizar, durante a fase de desenvolvimento de um sistema (ou algoritmo), os objectos, com especial interesse na verificação das transformações geométricas a que os polígonos são sujeitos ao longo do processo de visualização 3D.

5. Conclusões

Ao contemplar o processamento de cenas tridimensionais modeladas sob um formato de baixo nível, o NFF, e um formato de alto nível, o SIPP, o sistema FAROP constitui uma ferramenta extremamente útil na sua dupla função de apoio à rasterização e à visualização de objectos poligonais.

Na sua função de apoio à rasterização, o sistema FAROP disponibiliza as *interfaces* necessárias ao acoplamento de um motor de rasterização externo conforme a metodologia de teste pretendida. Um projectista poderá desejar desenvolver e, eventualmente, avaliar diferentes abordagens ao problema da rasterização, sem ter de se preocupar com a plataforma de execução. Neste caso, a pessoa integra no sistema a rotina, contendo o código do algoritmo de rasterização em causa, para posterior invocação. Num outro modo de teste, em que a plataforma de execução é relevante (e, muito provavelmente, diferente daquela em que o sistema de teste se encontra instalado), o FAROP produz um ficheiro de saída contendo a descrição da cena em Coordenadas de Ecrã o qual será posteriormente utilizado na máquina na qual se encontra o algoritmo de rasterização a ser ensaiado. Esta última metodologia de teste configura o quadro experimental em que o sistema foi, até agora, utilizado nos laboratórios das disciplinas de Computação Gráfica: ensaio dos algoritmos desenvolvidos pelos alunos com os ficheiros de teste produzidos pelo FAROP em Coordenadas de Ecrã.

No futuro, na posse das necessárias infraestruturas, pretende-se disponibilizar o sistema completo aos alunos de modo a que estes tenham acesso a todas as potencialidades do FAROP para modelação e visualização tridimensional.

Apêndice A. Ficheiros entrada/saída no sistema Farop

Neste Apêndice, apresentam-se os formatos dos ficheiros de entrada e saída para o sistema FAROP.

A.1. Ficheiro em formato NFF

v Definição de ponto de observação (Viewpoint);
from 1.02285 -3.17715 -2.17451 Localização do Viewpoint;
at -0.004103 -0.004103 0.216539 Localização do centro da imagem (Look at);
up -0.816497 -0.816497 0.816497 Vector que define a direcção "up";
angle 45 Ângulo de visualização (em graus);
hither 1 Distância do plano de "clipping" (frente) ao Viewpoint;
resolution 1024 1024 Resolução da imagem (em pixels x e y);
l 1.87607 -18.1239 -5.00042 Posição da fonte luminosa (x,y,z), por omissão assumida cor branca;
f 1 0.2 0.2 1 0 100000 0 0 Parâmetros da superfície (R,G,B,Kd,Ks,n,T,I) correspondendo a cor da superfície, coeficientes de reflexão difusa e especular, potência (coseno de "Phong") para modelação do grau de reflexão, grau de transparência e índice de refração (não usado);
p 3 Polígono com 3 vértices, seguido de coordenadas dos vértices
 -1 -1 1 3 x (x,y,z);
 -1 -0.9375 0.9375
 -0.9375 -1 0.9375
p 3 Polígono com 3 vértices, seguido de coordenadas dos vértices
 -0.9375 -0.9375 1
 -0.9375 -1 0.9375
 -1 -0.9375 0.9375
p 4 Polígono com 4 vértices, seguido de coordenadas dos vértices
 12 12 -0.5 4 x (x,y,z);
 -12 12 -0.5
 -12 -12 -0.5
 12 -12 -0.5
f 1 0.9 0.7 0.5 0.5 45.2776 0 0 Nova superfície;
s 0 0 0 0.5 Esfera com coordenadas do centro (x,y,z) e raio;
s 0.272166 0.272166 0.544331 0.166667
 ...

A.2. Ficheiro de saída em Coordenadas de Ecrã

O sistema permite exportar a cena sob a forma de um ficheiro contendo uma lista (*Flat File*) de todos os polígonos expressos nas suas coordenadas *Device Dependent* a serem visualizados (exclusivamente polígonos). Este ficheiro possui o seguinte formato:

```
512 512 512 128      Resolução máxima (x,y,z e cor);
120838              Número total de polígonos;
1                   Separador
3 100 50            Polígono com 3 vértices e coordenadas de Ymax/Ymin;
40 100 89 30 128 30 (x,y,z) e (RGB)
50 80 89 30 128 30
60 50 89 30 128 30
1
4 110 40            Polígono com 4 vértices e coordenadas de Ymax/Ymin;
40 40 89 30 128 30
50 110 89 30 128 30
60 50 89 30 128 30
55 60 89 30 128 30
1
127                Fim do ficheiro;
```


A.3. Informação de diagnóstico

Informação de diagnóstico exibida no ecrã durante a execução do programa.

NFF scene will be imported!
 Pipeline: Parsing NFF file, wait...
 Pipeline: Be aware that some SIPP and user defined view parameters will be override by NFF parameters!
 Pipeline: NFF background loaded
 Pipeline: NFF view parameters loaded
 Pipeline: NFF light loaded (6.00 6.00 6.00) (1.00 1.00 1.00)
 Pipeline: NFF light loaded (-6.00 6.00 6.00) (1.00 1.00 1.00)
 Pipeline: NFF surface with 1 polygons (0.30,0.60,0.30,5,1.00,0.85,0.70)
 Pipeline: NFF surface with 1 polygons (0.30,0.00,0.20,5,1.00,0.61,0.41)
 Pipeline: NFF surface with 145 polygons (0.30,0.00,0.20,5,1.00,0.61,0.41)
 Pipeline: NFF surface with 1 polygons (0.30,0.00,1.00,5,1.00,1.00,1.00)
 Pipeline: NFF surface with 145 polygons (0.30,0.00,1.00,5,1.00,1.00,1.00)
 Pipeline: NFF surface with 1 polygons (0.30,0.00,1.00,5,0.02,0.67,1.00)
 Pipeline: NFF surface with 145 polygons (0.30,0.00,1.00,5,0.02,0.67,1.00)
 Pipeline: NFF surface with 1 polygons (0.30,0.00,1.00,5,0.21,1.00,0.01)
 ...
 Pipeline: NFF surface with 145 polygons (0.30,0.00,1.00,5,1.00,0.26,0.75)
 Pipeline: Imported 129 surfaces with 9345 polygons, 0 spheres and 0 cones.
 Pipeline: NFF parser done
 Pipeline: Hither:1.00, Yon:15.00, Resolution(X,Y,Z,Color):1024,1024,1024,512
 Pipeline: Pipeline creating render data, wait...
 Timer started
 Pipeline: Image file to be created: 'pipe.ppm'
 Pipeline: Objects loaded: 3
 Pipeline: Surfaces loaded: 129
 Pipeline: Polygons loaded: 9345
 Pipeline: Polygons ready to be rendered by SIPP!
 Timer Stopped (Elapsed: 2.108s, User: 2.023s, System: 0.066s)
recursos;
 Pipeline: SIPP rendering, wait...
 Timer started
 Pipeline: Render Mode: Gouraud
 Pipeline: SIPP rasterization done!
 Timer Stopped (Elapsed: 112.693s, User: 108.058s, System: 1.053s)
 Pipeline: Successful End
 Pipeline: Polygons backfacing: 4864
 Pipeline: Polygons totally clipped: 39
 Pipeline: Polygons splited to triangles: 0
 Pipeline: Polygons rendered: 4442
 Pipeline: Releasing resources...!
 Pipeline: Release complete!

O ficheiro NFF "pipetest.nff" será processado;

Definição de fonte de luz, posição (x,y,z) e côr (RGB);

Características da superfície (Ka, Ks, Kd, n, R, G, B);

Total de objectos carregados;

Planos de "Clipping" em (z) e resolução (x,y,z) e côr;

*Início das operações de transformação geométrica;
 Ficheiro de Imagem (Output);*

Fim das operações de transformação geométrica e estatísticas de consumo de

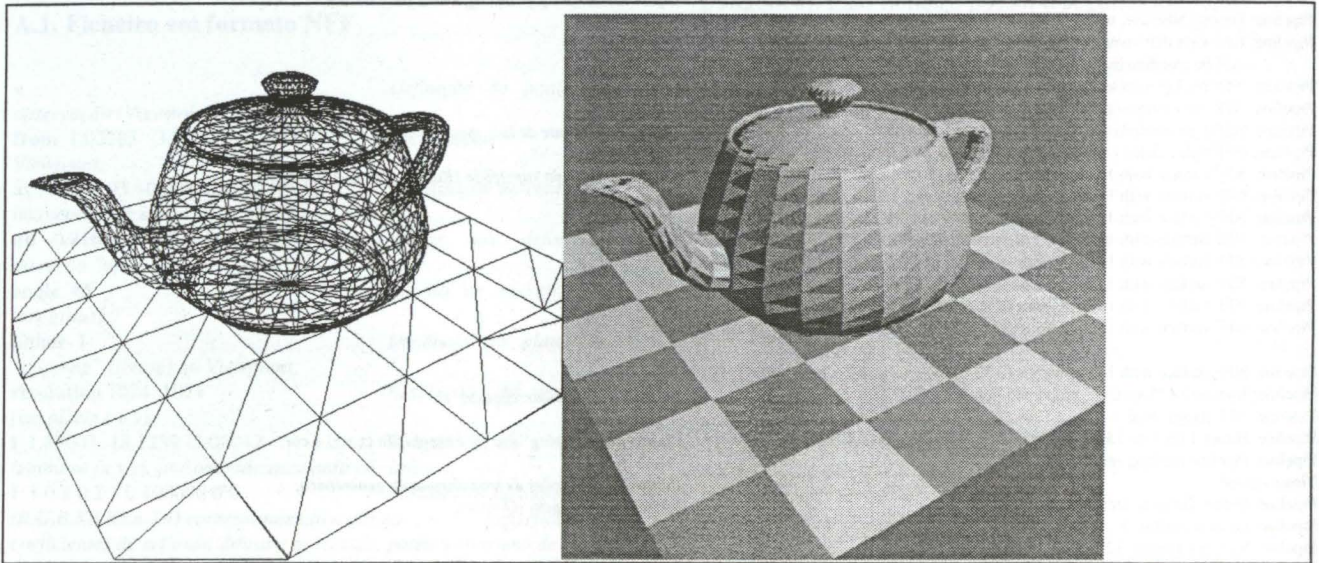
*Início da fase de rasterização;
 Método de sombreamento;
 Fim da fase de rasterização;
 Estatísticas, Fase de rasterização;*

*Total de polígonos removidos por "Culling";
 Total de polígonos totalmente removidos pela operação de "Clipping";
 Total de polígonos segmentados em triângulos;
 Total de polígonos rasterizados;*

Apêndice B. Exemplos ilustrativos das capacidades do Farop

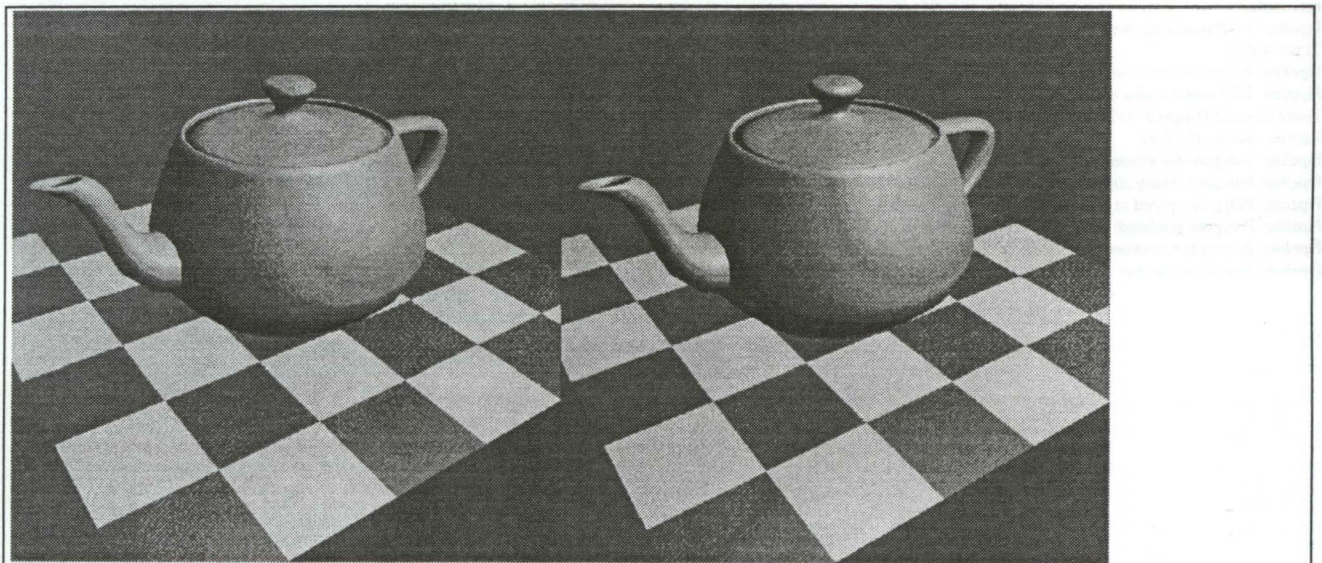
Neste Apêndice, mostram-se exemplos que exemplificam as potencialidades de representação do sistema FAROP.

B.1. Tipos de representação realista suportados pelo sistema



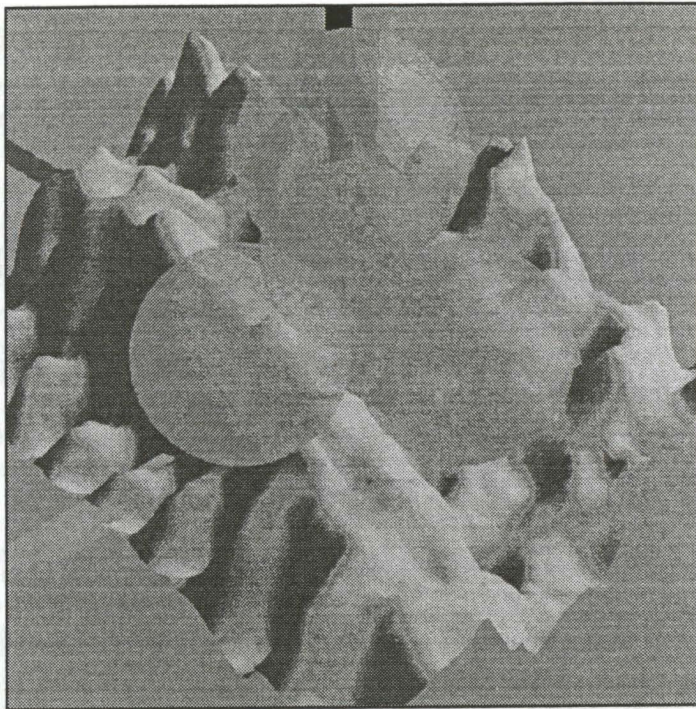
Teapot - modelo *Wire-Frame*

Teapot - Sombreamento constante



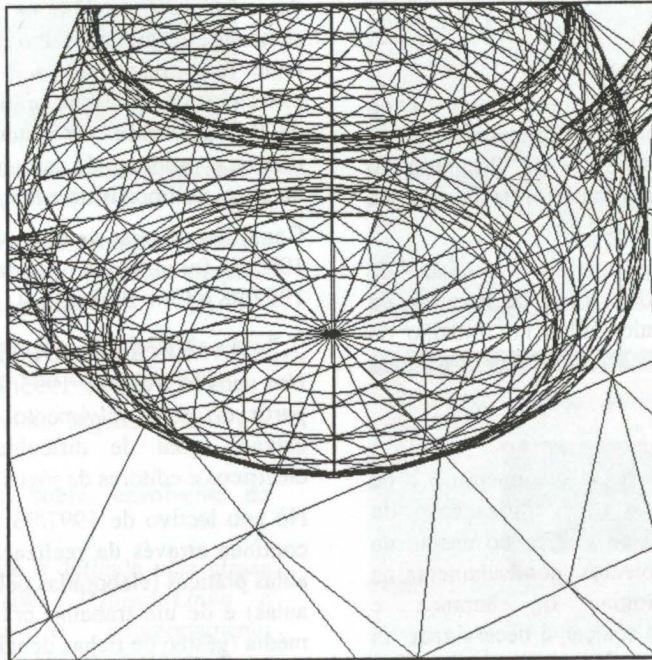
Teapot - Sombreamento *Gouraud*

Teapot - Sombreamento *Phong*



Cena *Mount* com esferas transparentes

B.2. Imagem recortada pelo algoritmo Sutherland-Hodgman



Cena *Teapot* recortada pelo algoritmo Cohen-Sutherland