



Lucas Caracas de Figueiredo

**Remoção de Superfícies Encobertas no Cálculo
de Área de Pintura em Modelos CAD**

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio.

Orientador: Prof. Waldemar Celes Filho

Rio de Janeiro
Agosto de 2017



Lucas Caracas de Figueiredo

Remoção de Superfícies Encobertas no Cálculo de Área de Pintura em Modelos CAD

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

Prof. Waldemar Celes Filho

Orientador

Departamento de Informática – PUC-Rio

Prof. Luiz Henrique de Figueiredo

IMPA

Prof. Marcelo Gattass

Departamento de Informática – PUC-Rio

Prof. Marcio da Silveira Carvalho

Coordenador Setorial do Centro Técnico Científico – PUC-Rio

Rio de Janeiro, 18 de Agosto de 2017

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Lucas Caracas de Figueiredo

Graduou-se no curso de Bacharelado em Ciência da Computação pela Universidade Federal do Maranhão (UFMA) em 14 de fevereiro de 2015.

Ficha Catalográfica

Figueiredo, Lucas Caracas de

Remoção de Superfícies Encobertas no Cálculo de Área de Pintura em Modelos CAD / Lucas Caracas de Figueiredo; orientador: Waldemar Celes Filho. – Rio de Janeiro: PUC-Rio, Departamento de Informática, 2017.

53 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2017.

Inclui bibliografia

1. Informática – Teses. 2. Campo de Distância Adaptativo;. 3. Geometria Sólida Construtiva;. 4. Área de Pintura;. 5. Modelos CAD.. I. Celes Filho, Waldemar. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

Agradecimentos

Em primeiro lugar, gostaria de agradecer especialmente a meu orientador Prof. Waldemar Celes, pelas conversas e ensinamentos, servindo como um guia muito importante nessa jornada. Gostaria de agradecê-lo pelo tempo dedicado na construção desta dissertação, pela disponibilidade e apoio. Me sinto muito afortunado por ter tido a oportunidade de trabalhar com ele durante esses anos.

Sou muito grato pela oportunidade de trabalhar com colegas muito competentes, que contribuíram de forma significativa no desenvolvimento desta dissertação. Eles foram responsáveis por proporcionar um ambiente divertido, estimulante e prazeroso, durante esses dois anos de pesquisa. Individualmente, agradeço ao Paulo Ivson, pela ajuda e disponibilidade durante a pesquisa, ao Eduardo Telles pelos ensinamentos passados, ao André Moreira pelo suporte e paciência, ao Wallas Santos pelo companheirismo e preocupação e ao Rafael Marques pela parceria.

Agradeço ao Instituto Tecgraf pelo apoio financeiro, em especial ao grupo CAE e ao Eduardo Thadeu, por terem me acolhido durante esse período. Gostaria de agradecer também aos meus amigos e companheiros de mestrado, Ruberth Barros, Alice Herrera, Rustam Mesquita, Vinícius Gama, pelos momentos de aprendizado compartilhados nas salas de aula.

Também sou grato aos membros da minha banca examinadora, Prof. Luiz Henrique de Figueiredo, Prof. Alberto Barbosa Raposo e Prof. Marcelo Gattass que disponibilizaram muito generosamente seu tempo e conhecimento para avaliar a minha dissertação.

Um agradecimento especial à Polyana, por cada momento de carinho, atenção, estímulo, amor, paciência e força. Aos meus pais, Severino e Luciana, pelo apoio e encorajamento; sem eles eu jamais estaria aqui.

Por último, mas não menos importante, gostaria de agradecer ao CNPq pelo apoio financeiro e à PUC-Rio pela bolsa de isenção das mensalidades do mestrado.

Resumo

Figueiredo, Lucas Caracas de ; Celes Filho, Waldemar. **Remoção de Superfícies Encobertas no Cálculo de Área de Pintura em Modelos CAD**. Rio de Janeiro, 2017. 53p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Sistemas CAD – Computer-Aided Design Systems – são muito utilizados nas diferentes etapas do ciclo de vida de um empreendimento de engenharia, como a elaboração do projeto conceitual, a construção da estrutura física e a operação da planta. A manutenção das instalações é uma tarefa de muita importância durante a operação, onde a pintura de equipamentos e estruturas é essencial. Estimar a área de pintura dos diferentes objetos possui um custo elevado se feito manualmente, com a utilização de trenas e lasers. Uma forma mais eficiente de calcular essas áreas é através do uso das ferramentas CAD. Entretanto, o processo de modelagem do modelo CAD, utilizando objetos paramétricos e malhas tridimensionais, insere superfícies que estão encobertas por outros objetos. Essas superfícies encobertas não são pintadas, e considerar suas áreas na orçamentação da pintura resulta em erros consideráveis. Portanto, o uso de um cálculo simples de todas as áreas de superfícies que compõem os objetos não é adequado. Com o objetivo de eliminar as superfícies escondidas do cálculo da área de pintura, este trabalho propõe uma abordagem baseada em campos de distância adaptativos juntamente com operações de geometria sólida construtiva. Primeiramente, as malhas passam por uma fase de pré-processamento, no qual são ajustadas de forma que cumpram com os requisitos necessários para a construção do campo de distância adaptativo, e em seguida os seus campos são calculados. Objetos parametrizados não necessitam dessa etapa pois já possuem um campo de distância implícito. Operações de geometria sólida construtiva foram então utilizadas para obter o campo da diferença e da interseção de cada objeto com a cena. De posse desses dados, foi desenvolvida uma fórmula que utiliza as áreas da diferença com a cena, da interseção e a área superficial de cada objeto para calcular a sua área de pintura. Em testes controlados, as áreas de pintura obtidas diferenciaram em no máximo 0,84% das áreas reais. Nos testes com modelos reais, foi obtido uma redução de até 38% da área estimada em relação a abordagem simplista de não tratar as superfícies ocultas.

Palavras-chave

Campo de Distância Adaptativo; Geometria Sólida Construtiva; Área de Pintura; Modelos CAD.

Abstract

Figueiredo, Lucas Caracas de ; Celes Filho, Waldemar (Advisor). **Hidden Surfaces Removal in Painting Area Calculation on CAD Models..** Rio de Janeiro, 2017. 53p. Dissertação de mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

CAD Systems – Computer-Aided Design Systems – are widely used in the different life cycle stages of an engineering enterprise, such as conceptual design, physical structure construction, and plant operation. The maintenance of the facility is a very important task during the operation, where painting the equipments and structures is essential. Estimating the painting area of the different objects has a high cost if done manually, using measuring tapes and lasers. A more efficiently way to calculate these areas is through the use of CAD tools. However, the modeling process of the CAD model, using parametric objects and three-dimensional meshes, inserts surfaces that are hidden by other objects. These hidden surfaces are not painted, and considering their areas in the painting budgeting leads to considerable errors. Therefore, the use of a simple calculation of all the surfaces areas that compose the objects is not adequate. With the objective of eliminating the hidden surfaces of the painting area computation, this work proposes an approach based on adaptive distance fields together with constructive solid geometry operations. Firstly, the meshes pass through a preprocessing phase, in which they are adjusted to fulfill the requirements for the adaptive distance field construction, and then their fields are computed. Parametrized objects do not need this step because they already have an implicit distance field. Constructive solid geometry operations were then used to obtain the difference and the intersection fields of each object with the scene. With this data, the painting areas are calculated considering the areas of the difference with the scene, the intersection and the surface area of each object. In controlled tests, the painting areas obtained differs of a maximum of 0.84% of the real areas. In tests with real models, a reduction of up to 38% of the estimated area was obtained in relation to the simplistic approach of not treating hidden surfaces.

Keywords

Adaptive Distance Fields; Constructive Solid Geometry; Painting Area; CAD Models.

Sumário

1	Introdução	14
2	Conceitos Básicos	18
2.1	Campos de Distância	18
2.2	Geometria Sólida Construtiva	23
2.3	Hierarquia de Volumes Envolventes	23
3	Cálculo da Área de Pintura Baseado em Campos de Distância	26
3.1	Visão Geral	26
3.2	Pré-processamento	27
3.3	Construção do Campo de Distância Adaptativo	29
3.4	Estimativa da Área de Pintura	33
4	Resultados e Discussão	38
4.1	Modelos Controlados	38
4.2	Modelos Reais	43
4.3	Discussão	46
5	Conclusão e Trabalhos Futuros	48
	Referências bibliográficas	50

Lista de figuras

Figura 1.1	Exemplos de superfícies encobertas: (a) cilindros verdes e torus circular transparente e verde com seus topos em contato, (b) caixa cinza transparente em contato com superfície azul abaixo e objeto cinza acima, (c) cilindro azul em interpenetração com objeto cinza.	16
Figura 2.1	Exemplo de campo de distância bidimensional em grade regular com sinal.	19
Figura 2.2	Letra 'R' representada por uma <i>quadtree</i> de três cores e um ADF. Imagens retiradas de (18).	20
Figura 2.3	Célula de um ADF com seguintes arestas destacadas: cima-direita, cima-frente, frente-direita, baixo-esquerda, baixo-atrás, atrás-esquerda.	22
Figura 2.4	Tipos de fendas que podem ocorrer na triangulação de grades adaptativas. Em (a) um tipo de fenda que não ocorre no algoritmo estendido do <i>SurfaceNets</i> . Em (b) um tipo de fenda que ocorre raramente e que pode ser evitada com uma etapa de pré-processamento. Imagens retiradas de (21).	22
Figura 2.5	Operações CSG. Imagens retiradas de (29).	23
Figura 2.6	A caixa envolvente de A e B não se sobrepõem, logo não necessita de um processamento mais complexo, enquanto as caixas C e D tem sobreposição, necessitando de mais processamento. Imagem retirada de (30).	24
Figura 2.7	Hierarquia de volumes envolventes com a cena dividida por AABBs em (a) e a respectiva árvore em (b).	25
Figura 3.1	Fluxograma de atividades.	26
Figura 3.2	Exemplos de geometrias que não possuem superfície fechada. Em (a) um cilindro vermelho sem tampa. Em (b) e (c) um torus circular que também não possui tampa. Em (d) uma válvula composta de várias geometrias, dentre elas uma semiesfera destacada com borda vermelha. Em (e) a semiesfera destacada em (d), expondo uma abertura na lateral.	28
Figura 3.3	Etapa de pré-processamento aplicada na semiesfera da Figura 3.2(e). Em (a) a semiesfera possui um ciclo de arestas que foram visitadas apenas uma vez; em (b) a triangulação em leque do ciclo; em (c) o resultado obtido da adição de (b) em (a) visualizado como <i>wireframe</i> ; em (d) o sólido obtido, com os novos triângulos em vermelho.	28
Figura 3.4	Campo de distância completo e com sinal de uma esfera.	29
Figura 3.5	Em (a) a visualização de uma <i>k-d tree</i> com triângulos, na qual existem folhas com mais de um triângulo, pois o plano divisor não consegue dividir os triângulos em grupos distintos. Em (b) a árvore de busca binária de (a).	31

Figura 3.6	Busca da menor distância entre P e o conjunto de triângulos na k - d tree da Figura 3.5.	32
Figura 3.7	Na malha, o ponto mais próximo de p é um vértice, porém os sinais dos produtos internos $d \cdot n_1$ e $d \cdot n_2$ são diferentes. Imagem retirada de (3).	32
Figura 3.8	Normal N do vértice calculada pela média das normais n_1 , n_2 e n_3 ponderadas pelos ângulos α_1 , α_2 e α_3 respectivamente. Imagem retirada de (3).	33
Figura 3.9	Em (a) a cena composta pelos objetos A, B e C, com a superfície de pintura de A em verde, as superfícies escondidas de A em vermelho e as superfícies escondidas de B e C em azul. Em (b) o objeto A separado da cena. Em (c) a diferença entre A e a cena. Em (d) a interseção entre A e a cena.	34
Figura 3.10	Cena com os objetos A e B, com a tolerância de acesso aplicada sobre os objetos, em tracejado, para a construção do ADF da interseção entre A e B.	35
Figura 3.11	Tolerância aplicada na cena da Figura 3.9 para o cálculo da área de pintura de A, ressaltando o aumento da área da interseção, em tracejado, com a cena.	36
Figura 3.12	Vértice p sendo deslocado no sentido do gradiente g em direção da superfície, com um valor equivalente a distância reconstruída d , chegando na superfície com a tolerância de acesso aplicada (tracejada). Ao descontar a tolerância aplicada t , a superfície anterior a aplicação da tolerância é alcançada.	36
Figura 4.1	Modelo com quatro esferas de raio igual a 0,5, apresentando calotas esféricas, de altura 0,2, em interpenetração. Em (a), a cena pode ser observada de um ponto de vista externo. Em (b), o interior da cena pode ser visualizado.	39
Figura 4.2	Em (a) é possível observar as diferenças entre cada objeto e a cena. Em (b), as interseções, entre cada objeto e a cena, são visualizadas.	40
Figura 4.3	Modelo com quatro caixas de lado 0,5, no qual as faces das caixas estão em contato. Em (a), o exterior da cena pode ser observado e, em (b), o seu interior.	41
Figura 4.4	Em (a), as diferenças entre cada caixa e a cena são visualizadas, e, em (b), as interseções.	41
Figura 4.5	Modelo com quatro cilindros de raio e altura iguais 0,5, apresentando contato entre as tampas dos cilindros. Em (a), o exterior da cena é visualizado, enquanto em (b), é possível observar seu interior.	42
Figura 4.6	Em (a), as diferenças entre cada objeto e a cena são notadas, já em (b), é possível observar as interseções entre cada objeto e a cena.	43
Figura 4.7	Em (a), uma válvula de um modelo real, composta por 23 objetos distintos, com uma área superficial de $0,3301\text{m}^2$. Em (b), uma válvula globo real, com 12 objetos diferentes e uma área superficial de $2,3870\text{m}^2$.	44

Figura 4.8 Em (a), um vaso, composto por 73 objetos distintos, com uma área superficial de $190,3764\text{m}^2$. Em (b), uma parte de uma unidade de compressão, com 373 objetos e uma área superficial de $39,5700\text{m}^2$.

45

Lista de tabelas

Tabela 4.1	Consumo de memória e desempenho obtidos na aplicação da abordagem proposta em modelos reais.	46
------------	--	----

Lista de Abreviaturas

AABB – *Axis Aligned Bounding Box*
ADF – *Adaptively sampled Distance Fields*
B-rep – *Boundary Representation*
BVH – *Bounding Volume Hierarchy*
CAD – *Computer-Aided Design*
CSG – *Constructive Solid Geometry*
GB – GigaBytes
MB – MegaBytes
RAM – *Random Access Memory*
voxblt – *Voxel Block Transfer*

Se tiver o hábito de fazer as coisas com alegria, raramente encontrará situações difíceis.

Baden-Powell.

1

Introdução

Sistemas CAD, *Computer-Aided Design Systems*, são ferramentas para a criação, visualização e manipulação de desenhos técnicos digitais, com curvas e figuras, e modelos tridimensionais através de superfícies e sólidos. O invento do CAD 3D é atribuído ao engenheiro francês Pierre Bézier, que na década de 1960 desenvolveu o sistema UNISURF(1), para facilitar a elaboração do projeto de diferentes ferramentas e peças para a indústria automobilística.

Atualmente, devido à sua versatilidade, as ferramentas CAD possuem uma ampla utilização pelo setor industrial, sobretudo nos setores extensivamente tecnológicos. O emprego desse tipo de ferramenta promove o aumento da produtividade dos profissionais envolvidos, principalmente, pela maior facilidade na organização e manipulação de dados, além da criação e visualização dos mesmos.

Quando voltados para empreendimentos de engenharia, esses sistemas, além de proporcionar uma melhor organização dos dados, permitem a execução de operações nos modelos CAD, tais como medir distância entre objetos e calcular a área de elementos, dentre outras. Essas características possibilitam a aplicação de ferramentas CAD nas diferentes etapas do ciclo de vida de um empreendimento, como a elaboração do projeto conceitual, construção e operação da planta (2).

Na etapa de elaboração do projeto, as ferramentas de modelagem são utilizadas para a criação conceitual do empreendimento. Durante a construção, facilitam o controle dos processos envolvidos na materialização do projeto conceitual. Nessa etapa, é importante que os modelos CAD sejam atualizados de acordo com a real montagem do empreendimento, para que haja uma fidelidade entre o modelo 3D e a construção, possibilitando a utilização deste modelo na próxima etapa.

Durante a operação, ferramentas CAD podem ser utilizados para acompanhar os indicadores de operação da planta, auxiliando na identificação e prevenção de falhas, permitindo uma rápida resposta dos operadores que estejam presentes na planta. Nessa etapa, a manutenção das instalações industriais é uma tarefa de elevada importância para garantir que o empreendimento opere de maneira segura e eficiente. A elaboração de um planejamento adequado de

manutenção pode ser um diferencial para o sucesso de um empreendimento. As ferramentas CAD auxiliam na elaboração desses planejamentos, aprimorando a utilização dos recursos disponíveis.

Instalações como plataformas petrolíferas, que permanecem bastante tempo sujeitas às fortes intempéries do alto mar, necessitam da constante manutenção de seus equipamentos e estruturas. Nesse sentido, uma etapa essencial da manutenção é a pintura com tinta anticorrosiva, adequada para cada tipo de equipamento. Estimar a quantidade de tinta necessária para pintar uma determinada zona de uma plataforma pode ser uma atividade de custo elevado, caso seja realizada de forma manual, na qual os objetos de interesse para a pintura são medidos por instrumentos como trenas e *lasers*, e então têm suas áreas calculadas.

Ferramentas CAD podem calcular essas áreas de forma mais eficiente, reduzindo de forma considerável o tempo empregado nesta tarefa, bem como os custos envolvidos. Contudo, a representação dessas plataformas através de objetos paramétricos e malhas 3D, no processo de modelagem, introduz uma série de superfícies que estão escondidas ou encobertas por outros elementos, através do contato e até mesmo da interpenetração com outros elementos, como pode ser observado na Figura 1.1.

Essas superfícies encobertas não são pintadas durante a manutenção. Dessa forma, calcular a área de pintura de um objeto através da sua representação 3D vai muito além do simples cálculo da área superficial individual de cada objeto. Em outras palavras, a superfície de pintura de uma cena corresponde a união das parcelas da superfície de cada objeto que não estão dentro, ou em contato, com as superfícies dos outros objetos. Assim, para obter a área de pintura de um objeto em relação a cena, basta encontrar a parte de sua superfície que pertence a fronteira da união dos objetos da cena.

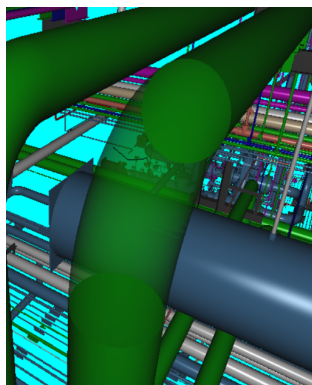
A representação de fronteira, *B-Rep*, do inglês *Boundary Representation*, dos objetos é uma alternativa para o cálculo da superfície de pintura, uma vez que é possível verificar, para cada triângulo de uma malha, a sua parcela pertencente à fronteira da união. Entretanto, essa operação possui um custo computacional elevado, pois, para obter a porção do triângulo que pertence à fronteira da união, ele é testado contra todos os outros triângulos da cena sendo que, se estiver totalmente dentro de outro objeto, é desconsiderado, se estiver totalmente fora é considerado por inteiro e, caso esteja em interseção com outro triângulo, ele é cortado, levando em consideração apenas a sua parte que está fora do objeto.

Além disso, a identificação de contato entre objetos, utilizando suas *B-Rep*, também é uma tarefa complicada, já que, quando em contato, um

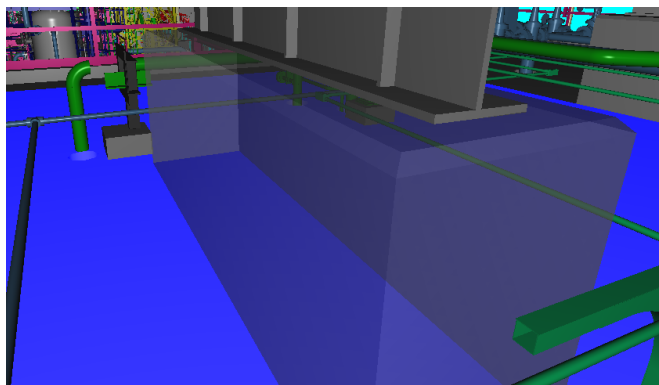
triângulo não está dentro de um objeto, e sim muito próximo. De fato, a representação da fronteira dos objetos, não é eficiente para a verificação de oclusão e contato. Campos de distância, por sua vez, representam uma alternativa mais adequada. Essa estrutura provê, de forma eficiente, a menor distância entre um ponto e a superfície que o mesmo representa (3), sendo muito empregadas na modelagem de objetos através da aplicação de geometria sólida construtiva (4).

Nesse contexto, o presente trabalho tem como objetivo desenvolver uma abordagem capaz de calcular a área de pintura de equipamentos e estruturas utilizando modelos CAD, desconsiderando as áreas das superfícies que estejam em contato ou escondidas, seguindo as etapas:

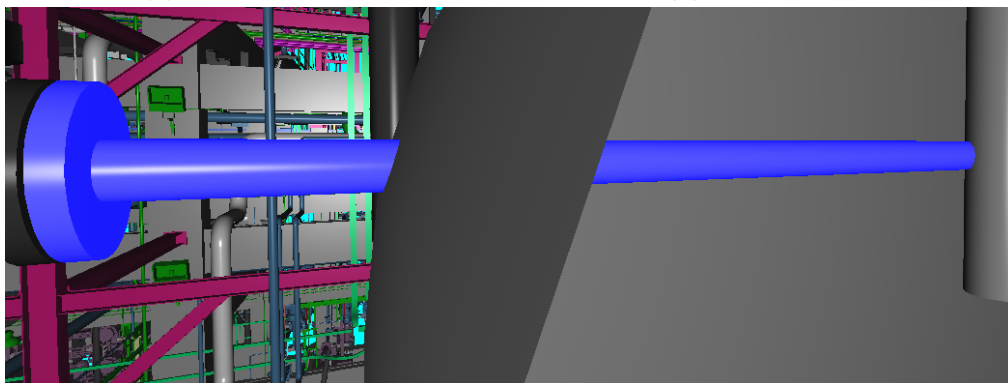
- (i) Obter os campos de distância dos objetos que terão suas área calculadas;
- (ii) Calcular a área de pintura do objeto levando em consideração apenas a parte de sua superfície que não esteja encoberta ou em contato com os demais objetos da cena.



(a)



(b)



(c)

Figura 1.1: Exemplos de superfícies encobertas: (a) cilindros verdes e torus circular transparente e verde com seus topos em contato, (b) caixa cinza transparente em contato com superfície azul abaixo e objeto cinza acima, (c) cilindro azul em interpenetração com objeto cinza.

Utilizando a abordagem proposta para computar a área de pintura de objetos em relação a cena, em casos controlados, foi observado que a área calculada pelo algoritmo e a área real de pintura diferenciaram em, no máximo, 0,84%, demonstrando a capacidade da abordagem de estimar com uma precisão razoável a área de pintura. Já em modelos reais, uma redução de até 38% da área da superfície de pintura foi alcançada, viabilizando uma economia sensível dos recursos empregados na manutenção de instalações industriais.

A presente dissertação está organizada em cinco capítulos. O primeiro capítulo consiste desta introdução. O segundo capítulo ('Conceitos Básico') traz o conceito de campo de distância, seus desafios, tipos e maneiras de construção, apresentando o algoritmo para a triangulação de um campo de distância adaptativo. Ao mesmo tempo, detalha as operações de geometria sólida construtiva aplicadas a campos de distância. Além disso, apresenta estruturas espaciais que permitem a divisão hierárquica da cena.

O terceiro capítulo ('Cálculo da Área de Pintura Baseado em Campos de Distância') descreve como são construídos os campos de distância para cada objeto, e apresenta como é estimada a área de pintura de cada objeto. Além disso, expõe algumas otimizações realizadas para acelerar o desempenho do cálculo para uma grande quantidade de objetos. Ainda, traz modificações feitas no algoritmo de triangulação de um campo de distância no sentido de calcular a sua área. O quarto capítulo ('Resultados e Discussões') apresenta e discute os resultados, tanto em casos controlados como em casos reais. Nesse capítulo é exposta a eficácia do método, além de comparar diversos resultados. Finalmente, o quinto capítulo ('Conclusões') apresenta as conclusões e propostas para trabalhos futuros.

2

Conceitos Básicos

Neste capítulo, primeiramente é apresentado o conceito de campos de distância, bem como as suas formas de construção e classificação, seguido do conceito de campo de distância adaptativo, suas vantagens e algoritmo para reconstrução da superfície do objeto representado. Em seguida, são revisadas as operações de geometria sólida construtiva e como são aplicadas a campos de distância. Por fim, são apresentadas estruturas de divisão hierárquica da cena e seus algoritmos de construção.

2.1

Campos de Distância

Campos de distância são uma estrutura de dados que utilizam uma nuvem de pontos para representar um objeto de interesse, sendo que, para cada ponto nessa nuvem é armazenado o valor da menor distância entre o ponto e a superfície do objeto. Dessa forma funcionam como uma função escalar, que fornece as distâncias mínimas para os pontos de uma região espacial, como a caixa envolvente do objeto (5). Essas distâncias podem ser com ou sem sinal, distinguindo o interior do exterior, como pode ser observado na Figura 2.1, na qual temos a distância zero nos pontos que estão sobre o objeto, valores em vermelho para pontos no interior e em verde para ponto no exterior. Em geral, os pontos dessa nuvem são dispostos em uma grade regular.

Esse tipo de estrutura apareceu primeiramente na literatura em 1966, em um artigo de processamento de imagem no qual Rosenfeld e Pfaltz (6) utilizaram a transformada de Chamfer para extrair o esqueleto de uma imagem, isto é, uma representação mínima da estrutura da imagem. Com o seu crescente estudo, passou a ser aplicada como uma forma de representação geométrica para modelos volumétricos. Kaufman et al. (7) apresentaram as vantagens desse tipo de representação, como o fato de permitirem a aplicação de operações de geometria sólida construtiva para modelagem.

Campos de distância armazenados em grades regulares tridimensionais são populares na computação gráfica, sendo utilizadas em diversas aplicações como a detecção de colisão (8, 9, 10, 11), aplicação de operações morfológicas (12, 13), modelagem de objetos (14, 15) e geração de malha (16, 17).

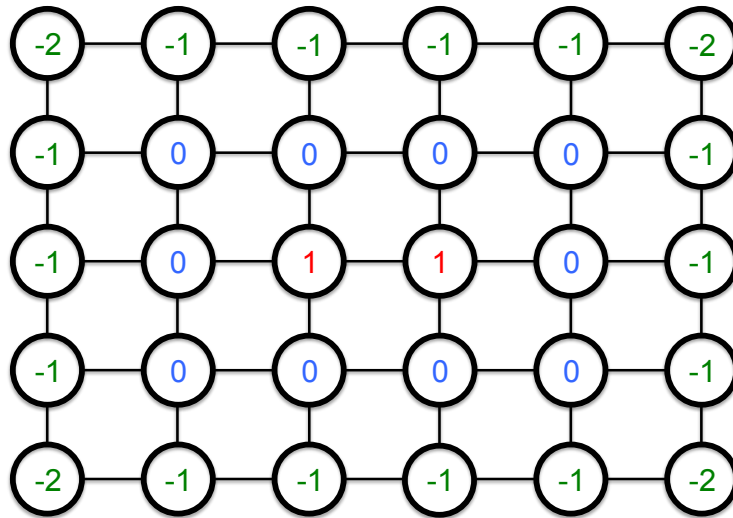


Figura 2.1: Exemplo de campo de distância bidimensional em grade regular com sinal.

Contudo, a representação de objetos em grades regulares possui algumas desvantagens, devido ao seu tamanho e resolução limitados. Como ressaltado em (3), para que seja possível capturar detalhes da geometria, uma alta resolução da grade é necessária, consumindo assim uma quantidade excessiva de memória.

2.1.1

Campos de Distância Adaptativos

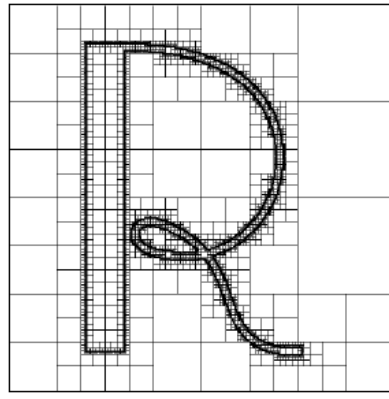
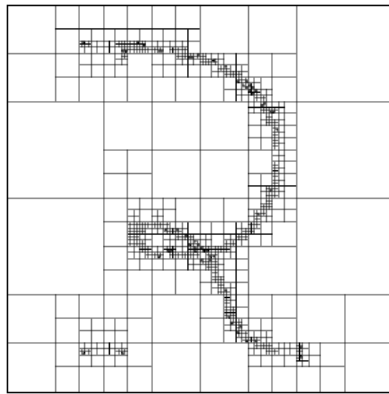
Com o intuito de resolver as limitações do uso de grades regulares, Frisken et al. (18) introduzem o conceito de campos de distância adaptativos ou ADF, do inglês *Adaptively sampled Distance Fields*. Essa estrutura possui semelhanças com a *octree* de três cores (19). Na *octree* de três cores as células são classificadas de acordo com a sua relação com a superfície do objeto de interesse como exteriores, interiores ou de fronteira, sendo que as células de fronteira sempre são subdivididas até um nível máximo predefinido. Já em um ADF as células de fronteira são subdivididas apenas quando a interpolação trilinear dos valores em seus vértices não garante a reconstrução da distância dentro de uma certa tolerância.

Na Figura 2.2, através da representação como *quadtrees*, é notado que um ADF necessita de uma quantidade de células bem menor para representar o objeto, pois utiliza células maiores para representar regiões planas do objeto. Além disso, também é possível observar que nas curvas são necessárias apenas algumas subdivisões, pois são representadas razoavelmente bem pela interpolação bilinear. Dessa forma, apenas os cantos necessitam de uma quantidade maior de subdivisões.

Sendo assim, um ADF consiste em um campo de distância armazenado numa estrutura de *octree*, na qual as células possuem uma variação no refinamento de acordo com a necessidade de representação de detalhes. Por esse motivo, esse tipo de estrutura acaba sendo mais indicada para a representação de campos de distância complexos de forma mais compacta.



(a) Letra 'R'

(b) *Quadtree* de três cores com 23.573 células.

(c) ADF com 1.713 células

Figura 2.2: Letra 'R' representada por uma *quadtree* de três cores e um ADF. Imagens retiradas de (18).

Dada uma função de distância, ADFs podem ser construídas seguindo duas abordagens (18): *top-down* ou *bottom-up*. Na abordagem *top-down*, primeiramente é calculado o valor das distâncias para os vértices da célula raiz, que cobre a região de interesse. Em seguida essa célula é subdividida recursivamente, até que o erro de reconstrução seja menor que uma tolerância previamente estabelecida, ou que uma profundidade máxima, previamente estabelecida, seja alcançada. Já na abordagem *bottom-up*, um campo de distância é construído sobre uma grade regular de resolução máxima e, em seguida, grupos de oito vizinhos são então unificados em uma célula maior caso nenhum dos oito tenham células filhas e os valores de distâncias dos seus vértices podem ser reconstruídos com um erro menor que uma tolerância previamente estabelecida.

Um ADF, de acordo com Figueiredo et al. (20), pode ser classificado em dois tipos: de fronteira ou global. O ADF é dito de fronteira quando somente as células de fronteira podem ser subdivididas. Nesse caso, apenas a reconstrução das distâncias próximas à isosuperfície de interesse é garantida com precisão. Já em um ADF global, todas as células podem ser subdivididas, garantindo a reconstrução da distância em todos os pontos do domínio com uma tolerância de erro previamente estabelecida.

2.1.2

Reconstrução da Superfície de um Campo de Distância Adaptativo

Friskén e Perry (21) propõem um algoritmo para a triangulação de um ADF. Esse algoritmo é uma extensão do *SurfaceNets* (22) para aplicação em grades adaptativas, que, a partir de consultas na ADF, produz uma malha topologicamente consistente e com triângulos próximos a triângulos equiláteros. O algoritmo consiste de três passos básicos:

- (i) Um vértice, inicialmente posicionado no centro da célula, é associado a cada célula folha que contém a isosuperfície de valor zero.
- (ii) Os vértices de células vizinhas são conectados para formar triângulos de acordo com os seguintes critérios: 1) um triângulo une os vértices de três células que compartilham uma mesma aresta; 2) um triângulo é associado a uma aresta apenas se essa intercepta a isosuperfície; 3) a orientação do triângulo pode ser definida através do sentido pelo qual a aresta cruza a isosuperfície e, 4) como as células possuem doze arestas, cada célula pode ter no máximo doze triângulos associados. No sentido de evitar redundâncias, apenas seis arestas são consideradas, indicadas na Figura 2.3.
- (iii) Após a formação de todos os triângulos, os vértices são movidos para a superfície, ou seja, um deslocamento com o valor da distância reconstruída na direção do gradiente. Em seguida, cada vértice é movido no sentido da média das posições dos seus vizinhos, visando aumentar a qualidade dos triângulos.

Além disso, Friskén e Perry (21) ainda demonstram como o algoritmo resolve o problema da produção de fendas nas malhas, tipicamente associadas com as grades adaptativas. A primeira forma de fenda, demonstrada em 2D na Figura 2.4(a), acontece quando a superfície é triangulada em células que possuem tamanhos diferentes, pois a posição interpolada do vértice produzida em uma célula pode não corresponder com a da produzida na célula vizinha.

Este tipo de fenda ocorre quando os vértices dos triângulos são gerados nas arestas ou faces da célula (23, 24, 25), portanto como o algoritmo proposto produz vértices nos centros das células, este tipo de fenda não acontece.

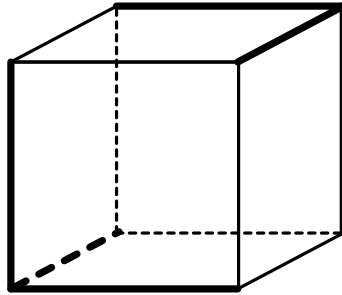


Figura 2.3: Célula de um ADF com seguintes arestas destacadas: cima-direita, cima-frente, frente-direita, baixo-esquerda, baixo-atrás, atrás-esquerda.

Entretanto os autores ressaltam que o algoritmo proposto produz um outro tipo de fenda, que raramente é produzida. Essa fenda é produzida quando o número de arestas que interceptam a isosuperfície é diferente entre dois conjuntos de células. Como pode ser observado na Figura 2.4(b), a aresta da célula acima intercepta a isosuperfície duas vezes, enquanto as arestas das suas células vizinhas interceptam a isosuperfície uma vez.

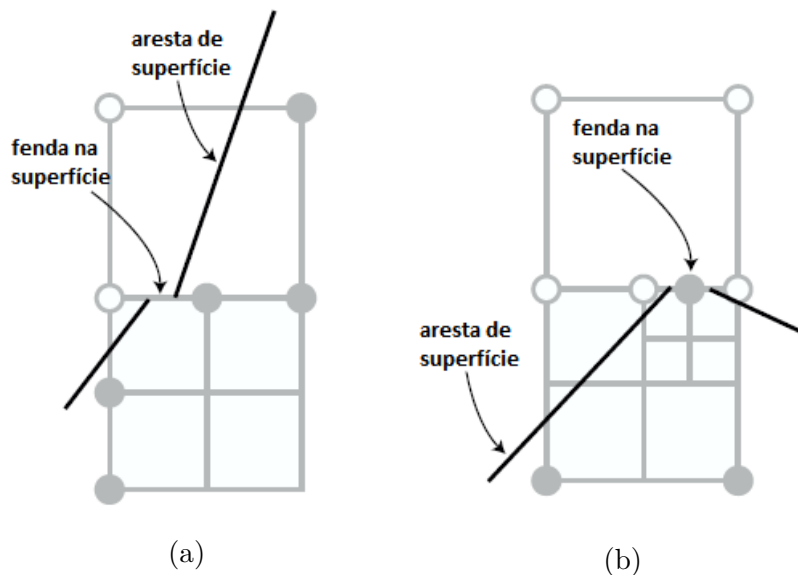


Figura 2.4: Tipos de fendas que podem ocorrer na triangulação de grades adaptativas. Em (a) um tipo de fenda que não ocorre no algoritmo estendido do *SurfaceNets*. Em (b) um tipo de fenda que ocorre raramente e que pode ser evitada com uma etapa de pré-processamento. Imagens retiradas de (21).

Para evitar esse tipo de fenda, uma etapa de pré-processamento é necessária. Nesta etapa é comparado o número de vezes que uma célula folha de fronteira intercepta a isosuperfície e dos seus vizinhos, caso sejam diferentes,

a célula é subdividida utilizando os valores das faces dos seus vizinhos. Na Figura 2.4(b) a célula acima seria subdividida em dois níveis durante esse pré-processamento.

2.2

Geometria Sólida Construtiva

Geometria sólida construtiva, ou CSG do inglês *Constructive Solid Geometry*(26), é um poderoso paradigma de modelagem que possibilita, a partir de formas geométricas simples, como cilindros, cubos, esferas, dentre outros, a composição de outras formas geométricas mais complexas. Essas geometrias podem ser adicionadas, subtraídas ou intersectadas, criando uma grande variedade de sólidos.

Essas operações volumétricas de CSG (27) podem ser aplicadas em campos de dados binários, ou escalares, através de operações nas quais o novo valor em um ponto da grade pode ser definido através da combinação dos valores presentes neste ponto em cada um dos campos. Esse tipo de operação é conhecida como *voxblt* (28), do inglês *Voxel Block Transfer*.

Frisken e Perry (29), de acordo com a convenção estabelecida por Frisken et al. (18), na qual valores positivos de distância representam pontos no interior do objeto e valores negativos pontos no exterior, apresentaram as operações *booleanas* através de mínimos e máximos, sendo a união dada por $dist(A \cup B) = \max(dist(A), dist(B))$, a diferença por $dist(A - B) = \min(dist(A), -dist(B))$ e a interseção por $dist(A \cap B) = \min(dist(A), dist(B))$. Essas operações estão ilustradas na Figura 2.5.

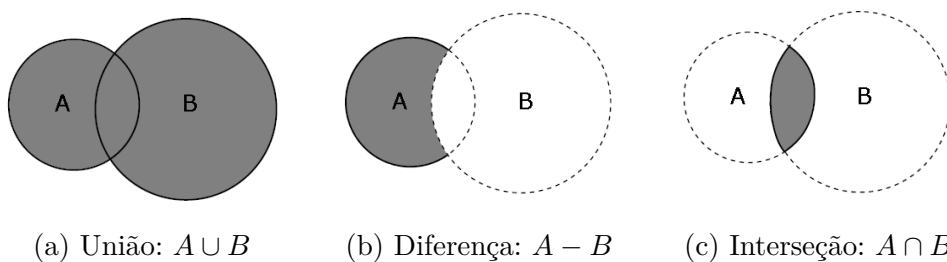


Figura 2.5: Operações CSG. Imagens retiradas de (29).

2.3

Hierarquia de Volumes Envolventes

Um volume envolvente é uma estrutura que, como o próprio nome sugere, envolve um ou mais objetos com o intuito de simplificar sua representação. Esse tipo de estrutura fornece testes de interseção de maneira mais eficiente do que os objetos que envolvem, como observado na Figura 2.6. Assim, um

processamento mais complexo pode ser realizado apenas com os objetos que possuem seus volumes envolventes sobrepostos.

Um tipo muito comum de volumes envolventes são as caixas alinhadas com os eixos, ou AABBs do inglês *Axis Aligned Bounding Boxes*. Essas caixas são retangulares e possuem as normais de suas faces orientadas paralelamente aos eixos do sistema de coordenadas utilizado. Sua principal característica é o rápido teste de sobreposição, além de consumir uma quantidade mínima de memória pois pode ser representada apenas com os pontos máximo e mínimo. Dessa forma, duas AABBs, a e b , possuem sobreposição apenas se, em todos os seus eixos, o ponto máximo de a for maior ou igual ao mínimo de b , e o mínimo de a for menor ou igual ao máximo de b .

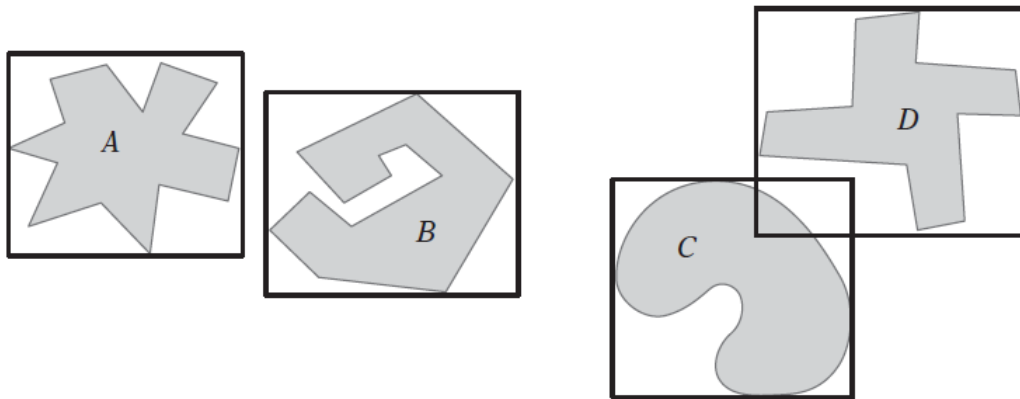


Figura 2.6: A caixa envolvente de A e B não se sobrepõem, logo não necessita de um processamento mais complexo, enquanto as caixas C e D tem sobreposição, necessitando de mais processamento. Imagem retirada de (30).

Contudo, apenas a simplificação de objetos através de seus volumes envolventes não diminui a quantidade de testes que serão realizados quando um objeto é processado em relação à cena inteira. Para que a quantidade de testes de sobreposição sejam reduzidas, aumentando a eficiência do processamento, esses volumes podem ser organizados em uma árvore hierárquica, conhecida como hierarquia de volumes envolventes (Figura 2.7), ou BVH do inglês *Bounding Volume Hierarchy* (30). Essas árvores, geralmente utilizadas para acelerar a detecção de colisão entre um objeto e a cena, facilitam a obtenção dos objetos que podem ter alguma interseção com um objeto de interesse.

Para construir uma BVH de AABBs, Bergen (31) enumera os seguintes passos:

- (i) Inserir todos os objetos no nó raiz;
- (ii) Computa a AABB para os objetos de um nó;

- (iii) Divide a AABB em seu maior eixo;
- (iv) As primitivas são então divididas de acordo com o centro de sua AABB;
- (v) Recursivamente, executa os passos (ii), (iii) e (iv), até que nenhuma divisão seja mais possível.

Ao executar esses passos na cena apresentada na Figura 2.6, a estrutura hierárquica da Figura 2.7 é obtida, tendo como nó raiz, a caixa envolvente verde, como nós intermediários, as caixas laranjas, e como nós folhas, as caixas azuis. Assim, para encontrar os outros objetos que D , por exemplo, pode interceptar, basta ir descendo na árvore pelos ramos em que a AABB do nó e a AABB de D se sobreponham. No final todos os nós folhas que estão próximos à D são encontrados. No exemplo, apenas a folha C .

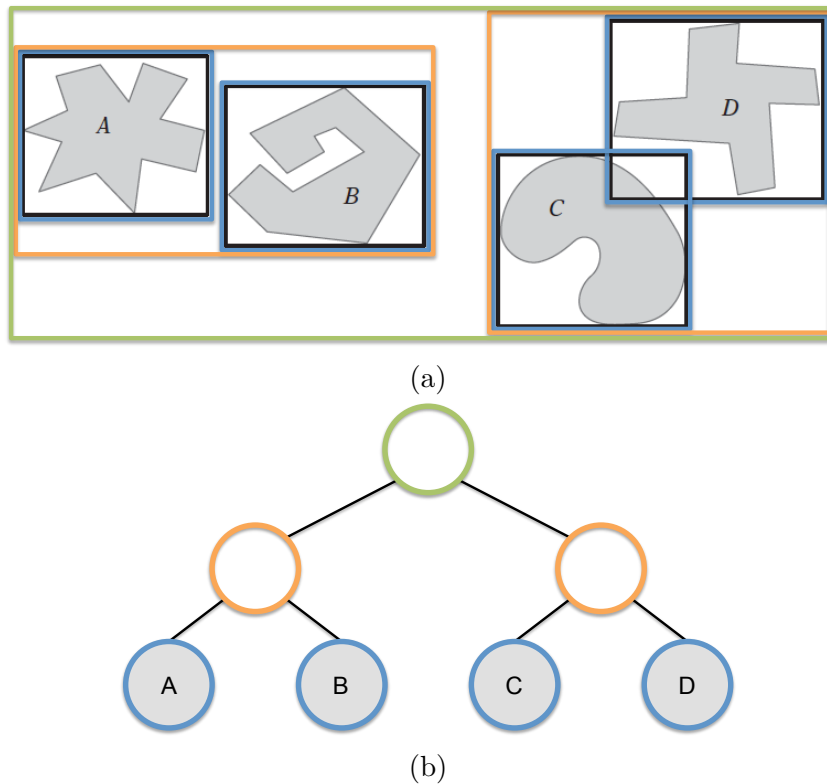


Figura 2.7: Hierarquia de volumes envolventes com a cena dividida por AABBs em (a) e a respectiva árvore em (b).

3

Cálculo da Área de Pintura Baseado em Campos de Distância

Neste capítulo é exposto, de forma detalhada, as etapas propostas para a eliminação das superfícies encobertas no cálculo da área de pintura, que resulta em estimativas com erros pequenos.

3.1

Visão Geral

Modelos CAD 3D possuem uma variedade de elementos representados através de malhas de triângulos e superfícies parametrizadas, como esferas, caixas, cilindros e cones. Durante o processo de criação desses modelos, superfícies encobertas, as quais não são pintadas, são produzidas. Com o objetivo de eliminar as superfícies encobertas do cálculo da área de pintura de objetos nesses modelos CAD, foi desenvolvida uma abordagem baseada em campos de distância adaptativos e operações CSG.

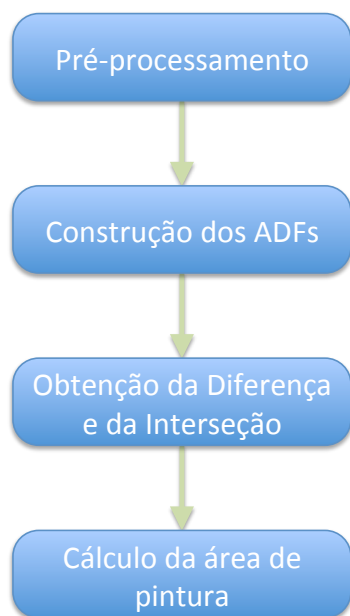


Figura 3.1: Fluxograma de atividades.

A abordagem consiste de quatro etapas, como ilustrado no fluxograma de atividades apresentado na Figura 3.1. A primeira etapa é o pré-processamento das malhas de triângulos para que elas cumpram os requisitos necessários para o cálculo dos ADFs. Esses requisitos são detalhados na Seção 3.2.

A segunda etapa, construção do campo de distância adaptativo com sinal, é aplicada, assim como a primeira, somente às malhas de triângulos, pois os objetos parametrizados possuem um campo de distância implícito, ou seja, dado um ponto, é possível calcular analiticamente, através da sua função implícita, a sua distância com sinal para a superfície. Portanto, esses objetos não precisam da construção do seu campo para processamentos futuros.

Na terceira etapa, os campos de distância equivalentes à diferença e à interseção do objeto com a cena são calculados, operações necessárias para o cálculo da área de pintura. Por fim, na quarta etapa, a área dos campos da diferença e da interseção são calculados, para em seguida a área de pintura do objeto ser obtida, como será exposto na Seção 3.4.

A terceira e a quarta etapas são aplicadas para cada objeto individualmente, obtendo assim a área de pintura dos objetos de forma gradativa, evitando um consumo de memória excessivo, pois, essa abordagem é aplicada em modelos CAD massivos, nos quais a quantidade de objetos que tem suas áreas de pintura calculadas é relativamente alta.

3.2

Pré-processamento

Uma das formas frequentemente utilizadas para representar objetos tridimensionais são as malhas de triângulos. Entretanto, para que seja possível calcular o campo de distância com sinal dessas malhas, elas precisam representar uma superfície fechada que seja *2-manifold*. Na prática, é possível impor as três condições descritas por Hoffmann (26), que são: a malha não pode possuir auto-interseção, suas arestas devem ser compartilhadas por exatamente dois triângulos, e todos os triângulos que incidem em um vértice devem formar um único ciclo ao redor do mesmo.

Em modelos reais, nem sempre essas malhas obedecem a todas as condições necessárias para a construção do ADF. Geometrias que não representam superfícies fechadas são frequentemente encontradas, como ilustrado na Figura 3.2. Assim, apesar de preencherem o requisito de não possuírem auto-interseção, apresentam arestas que pertencem a apenas um triângulo, além de terem vértices os quais os triângulos incidentes não formam um ciclo ao seu redor.

Para adequar as malhas de triângulos para que atendam a todos os requisitos para a construção dos ADFs, foi elaborada uma etapa de pré-processamento, exemplificada na Figura 3.3, na qual as aberturas são identificadas e fechadas. Primeiramente, nessa etapa, as arestas da malha são percorridas e, para cada aresta, é contado o número de vezes que são visitadas,

isto é, o número de triângulos adjacentes.

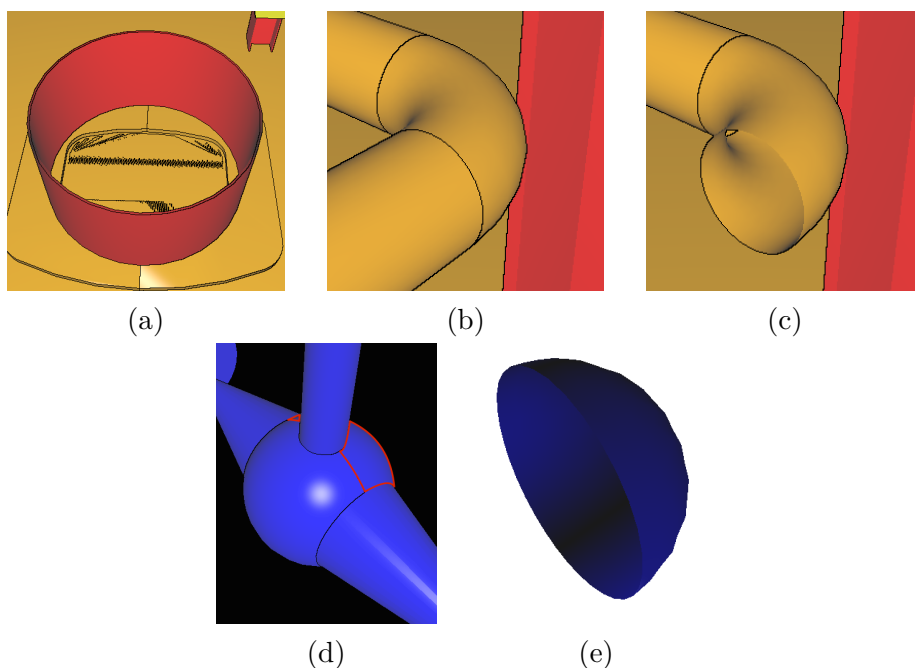


Figura 3.2: Exemplos de geometrias que não possuem superfície fechada. Em (a) um cilindro vermelho sem tampa. Em (b) e (c) um torus circular que também não possui tampa. Em (d) uma válvula composta de várias geometrias, dentre elas uma semiesfera destacada com borda vermelha. Em (e) a semiesfera destacada em (d), expondo uma abertura na lateral.

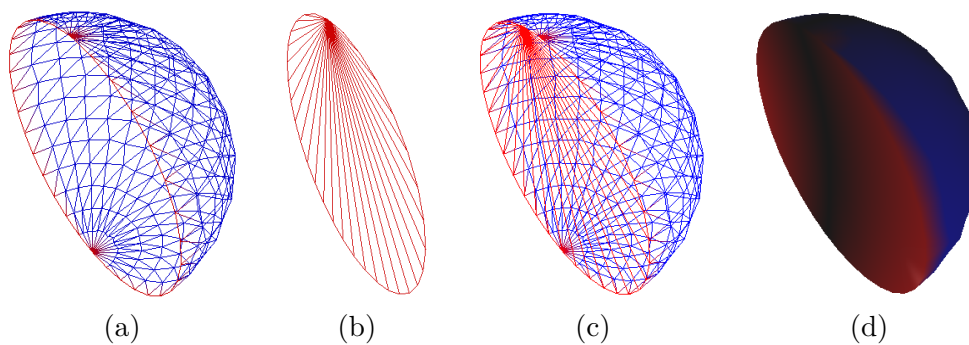


Figura 3.3: Etapa de pré-processamento aplicada na semiesfera da Figura 3.2(e). Em (a) a semiesfera possui um ciclo de arestas que foram visitadas apenas uma vez; em (b) a triangulação em leque do ciclo; em (c) o resultado obtido da adição de (b) em (a) visualizado como *wireframe*; em (d) o sólido obtido, com os novos triângulos em vermelho.

De posse dessa informação, as aberturas podem ser identificadas através de ciclos de arestas que são visitadas apenas uma vez. Caso a malha não possua nenhuma aresta que seja visitada apenas uma vez, ela segue para a próxima etapa. Caso contrário, a malha tem suas aberturas trianguladas, em forma de leque, na qual um vértice é selecionado e então todas as diagonais desse vértice são traçadas. Após essa triangulação, as aberturas são fechadas, de

forma que ao término desta etapa, todas as malhas preenchem os requisitos para a construção de seus ADFs.

3.3

Construção do Campo de Distância Adaptativo

Para construir o campo de distância adaptativo, ilustrado na Figura 3.4, Frisken e Perry (21) ressaltam que as abordagens *top-down* e *bottom-up*, descritas em (18), possuem limitações. Uma quantidade de memória excessiva é necessária para construir ADFs de alta resolução, além de uma grande quantidade de cálculos de distância, quando a abordagem *bottom-up* é utilizada. Já na abordagem *top-down* ocorrem uma grande quantidade de cálculos redundantes, tanto para obter a distância quanto para reconstruí-la.

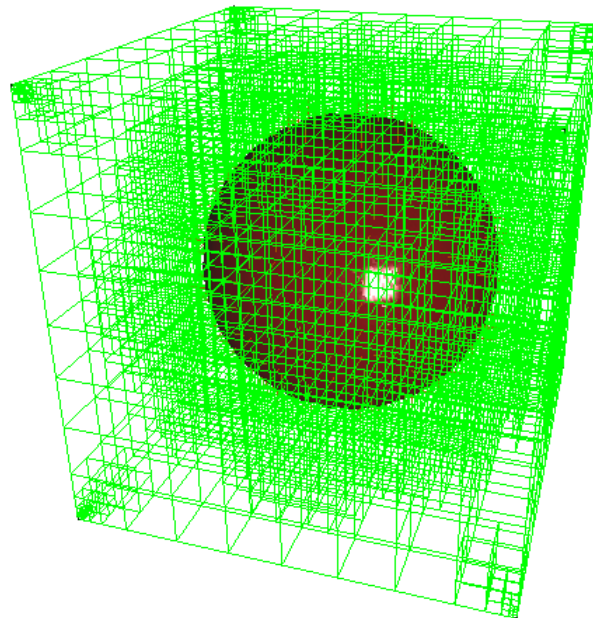


Figura 3.4: Campo de distância completo e com sinal de uma esfera.

Nesse sentido, inspirado na construção em mosaico (21), foi utilizado uma abordagem híbrida, na qual primeiramente é construído um campo de distância adaptativo utilizando a abordagem *bottom-up*, contudo essa abordagem é utilizada para criar um ADF com a metade da profundidade máxima previamente estabelecida, evitando uma grade inicial de alta resolução e consequentemente suas desvantagens associadas. A partir desse ponto intermediário, a abordagem *top-down* foi utilizada, tendo sua quantidade de cálculos redundantes amenizados, uma vez que começa a ser aplicada a partir da metade da profundidade máxima suportada.

Durante a construção do campo de distância adaptativo, uma célula necessita de subdivisão se a reconstrução da distância através da interpolação trilinear dentro dessa célula apresenta um erro maior que uma tolerância preestabelecida. Para definir se uma célula deve ser subdividida, geralmente, os pontos testados são o centro da célula, o centro das faces e das arestas. Na técnica proposta, apenas os pontos correspondentes aos centros de cada face e do centro da célula são testados, pois foi observado um aumento na velocidade da construção dos campos, sem que os cálculos de área de pintura fossem afetados de maneira significativa.

O campo de distância calculado é um campo global, como percebido na Figura 3.4, garantindo assim uma maior precisão dos cálculos que são executados, pois assegura que a distância, em qualquer ponto do domínio, pode ser obtida dentro de um erro tolerado, ao contrário do campo de fronteira, no qual apenas as distâncias em pontos próximos à superfície são reconstruídas dentro desse erro tolerado.

3.3.1

Malha de Triângulos

A construção de um campo de distância que representa uma geometria descrita através de uma malha de triângulos requer que a menor distância de um ponto para a malha de triângulos seja calculada. Esse cálculo pode ser feito de forma trivial, na qual a distância é obtida visitando-se todos os triângulos e escolhendo a menor distância encontrada. Contudo, como essa operação é executada diversas vezes, utilizar uma estrutura que possa acelerar esse processo diminui de forma considerável o custo envolvido (32).

Uma estrutura conhecida, a *k-d tree* (33), foi então utilizada, reduzindo a complexidade do cálculo da distância entre um ponto e a malha de triângulos de $O(n)$ para $O(\log n)$, onde n é a quantidade de triângulos encontrados na malha. Neste trabalho a *k-d tree* é uma árvore binária de busca que divide o espaço em uma dimensão por vez, sendo que o eixo escolhido para divisão é o de maior comprimento, considerando uma caixa alinhada que envolve todos os triângulos.

Após a escolha do eixo em que será realizada a divisão, os triângulos são ordenados de acordo com a posição do seu centro projetada no eixo, o triângulo correspondente à mediana é armazenado no nó e o seu centro escolhido como plano divisor. Em seguida, os outros triângulos são separados em dois grupos, de acordo com a posição do seu centro em relação ao plano escolhido, sendo que, caso algum dos vértices do triângulo esteja do outro lado do plano divisor, o triângulo é inserido nos dois lados. A divisão continua recursivamente até que

não seja mais possível dividir o conjunto no eixo escolhido, isto é, até que a quantidade de triângulos no nó for igual a um, ou o plano divisor escolhido não separar os triângulos em dois grupos distintos. Um exemplo dessa divisão espacial pode ser observado em 2D na Figura 3.5.

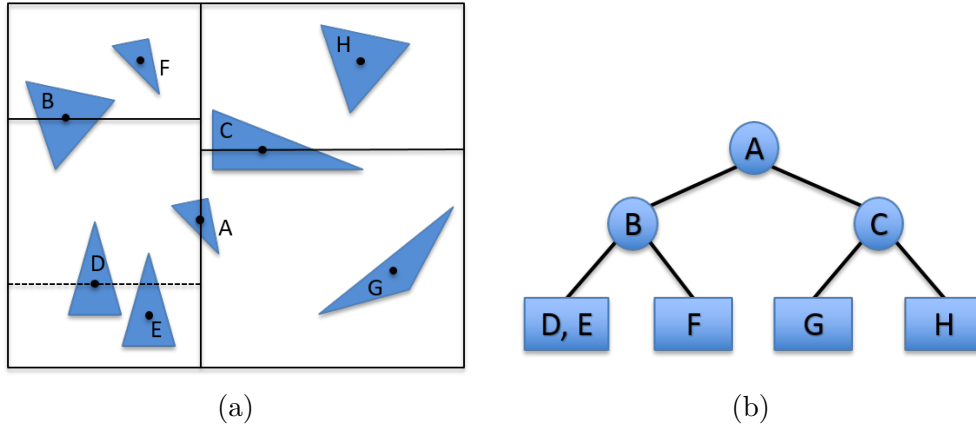


Figura 3.5: Em (a) a visualização de uma k - d tree com triângulos, na qual existem folhas com mais de um triângulo, pois o plano divisor não consegue dividir os triângulos em grupos distintos. Em (b) a árvore de busca binária de (a).

De posse da k - d tree dos triângulos da malha, para encontrar a menor distância entre um ponto e a malha, o algoritmo de busca pelo vizinho mais próximo (34) foi utilizado. No algoritmo, a k - d tree é percorrida, descendo pelo lado da árvore em que o ponto esteja. Ao chegar em um nó folha, a distância é então calculada para todos os triângulos e a menor é armazenada. Enquanto a recursão é desfeita, se a distância para o nó corrente é menor que a distância encontrada até o momento, então ela se torna a menor distância. Em seguida, é verificado se a menor distância encontrada é maior que a distância para o plano divisor do nó. Em caso afirmativo, o outro lado da árvore também deve ser visitado. Ao finalizar esse processo para o nó raiz, a menor distância é obtida. Essa busca é ilustrada em 2D na Figura 3.6.

No início a árvore é percorrida até que a folha G é alcançada, a distância d_1 é computada e armazenada. Ao retornar para C , é verificado que as distâncias do ponto para C e para o eixo divisor são maiores que d_1 , logo d_1 continua sendo a menor distância encontrada e a outra subárvore de C não precisa ser verificada. Chegando em A é percebido que d_2 é menor que d_1 , assim d_2 passa a ser a menor distância encontrada. Além disso, d_2 é maior que a distância do ponto para o plano divisor de A , então a outra subárvore deve ser verificada. Ao descer na outra subárvore de A , a folha D, E é alcançada e, como d_3 é menor que d_2 , d_3 passa a ser a menor distância. Ao desfazer a recursão voltando para B , a menor distância continua sendo d_3 e a outra

subárvore não precisa ser analisada. Chegando novamente em A , o processo foi finalizado para o nó raiz, sendo d_3 a menor distância entre P e os triângulos da k - d tree.

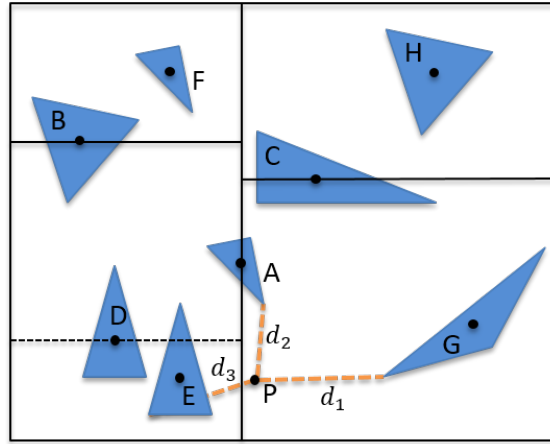


Figura 3.6: Busca da menor distância entre P e o conjunto de triângulos na k - d tree da Figura 3.5.

Outro ponto importante na construção de um campo de distância a partir de uma malha de triângulos é o cálculo do sinal. Determinar o sinal da distância entre um ponto p e um triângulo pode ser feito de forma trivial através do produto escalar entre a normal do triângulo e o vetor da direção do ponto no triângulo mais próximo de p ao ponto p . Contudo, nas malhas de triângulos, as normais não são bem definidas nas arestas e vértices. A utilização da normal do triângulo que está sendo testado pode acarretar erros, pois existe a possibilidade de um ponto ter a mesma distância, porém com sinais diferentes, para mais de um triângulo, como pode ser observado na Figura 3.7.

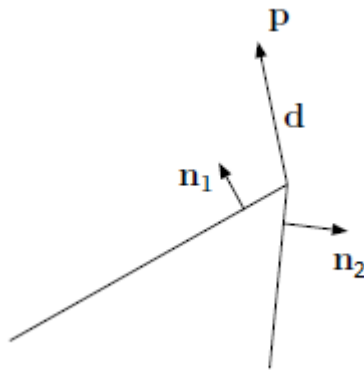


Figura 3.7: Na malha, o ponto mais próximo de p é um vértice, porém os sinais dos produtos internos $d \cdot n_1$ e $d \cdot n_2$ são diferentes. Imagem retirada de (3).

Aanæs and Bærentzen (35, 36) utilizam pseudo-normais, que são calculadas através da média ponderada pelos ângulos (37): a normal em um vértice

é definida pela soma das normais dos triângulos ponderadas pelo ângulo que as arestas dos triângulos formam naquele vértice (Figura 3.8). Esse conceito é aplicado também nas arestas, porém, como ambas as normais das faces são ponderadas por π , na prática funciona como a média entre as normais.

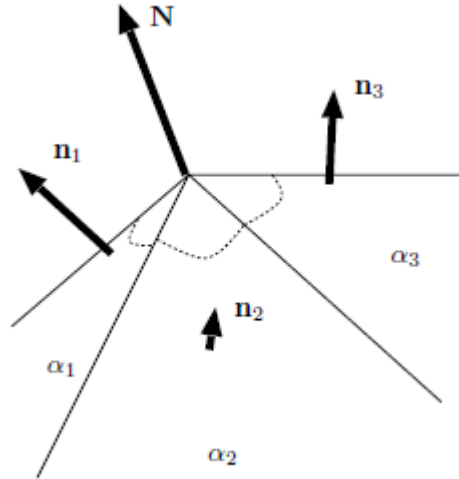


Figura 3.8: Normal N do vértice calculada pela média das normais n_1 , n_2 e n_3 ponderadas pelos ângulos α_1 , α_2 e α_3 respectivamente. Imagem retirada de (3).

3.4

Estimativa da Área de Pintura

Com o objetivo de estimar a área de pintura de um objeto A em relação à cena, foi observado que existe uma relação entre a área original do objeto, a área da interseção entre A e a cena, e a área da diferença entre A e a cena. Como ilustrado na Figura 3.9, a área total da superfície do objeto A consiste da sua área de pintura, em verde, mais a área da superfície escondida de A , em vermelho, ou seja:

$$A_o = A_p + A_{eo} \quad (3-1)$$

onde A_o a área total do objeto, A_p é a área de pintura e A_{eo} é a área da superfície encoberta do objeto.

É possível notar também que a área da superfície da diferença entre A e a cena é composta da área de pintura de A , em verde, somada das áreas das superfícies encobertas dos objetos em interseção, B e C , em azul. Dessa forma a área da diferença entre A e a cena é dada por:

$$A_d = A_p + A_{ec} \quad (3-2)$$

onde A_d a área da diferença e A_{ec} é a área das superfícies encobertas da cena. Além disso, é percebido que a área da superfície da interseção entre A e a

cena é a soma entre a área das superfícies encobertas de A e das superfícies encobertas da cena, B e C , isto é:

$$A_i = A_{eo} + A_{ec} \quad (3-3)$$

onde A_i a área de interseção entre A e a cena. Assim, a área de pintura de A pode ser estimada:

$$\begin{aligned} A_o + A_d - A_i &= (A_p + A_{eo}) + (A_p + A_{ec}) - (A_{eo} + A_{ec}) \\ &= 2(A_p) \\ \therefore A_p &= \frac{A_o + A_d - A_i}{2} \end{aligned} \quad (3-4)$$

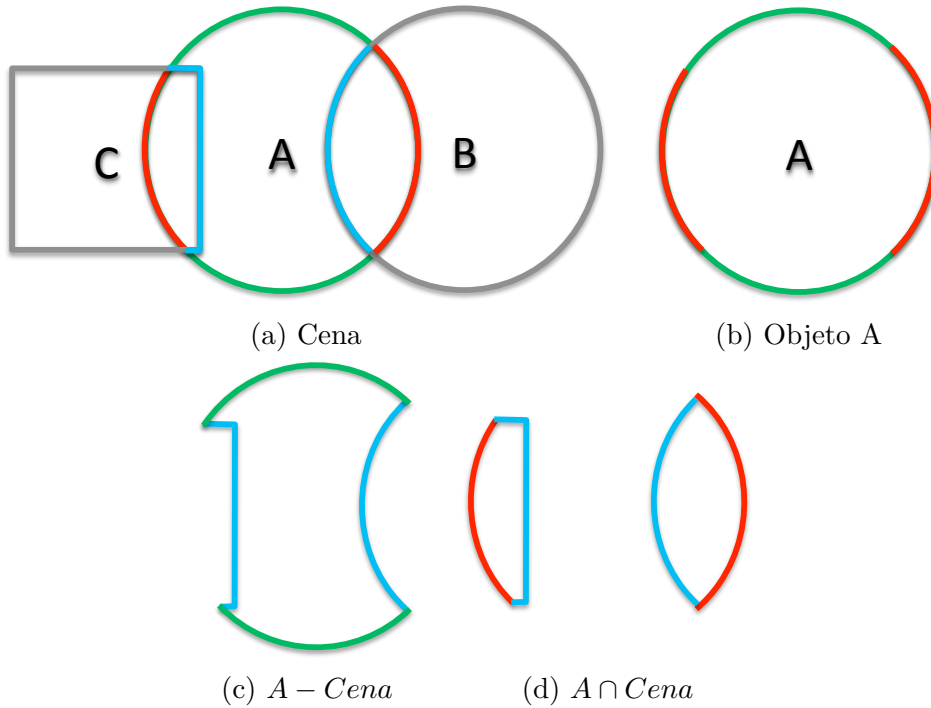


Figura 3.9: Em (a) a cena composta pelos objetos A , B e C , com a superfície de pintura de A em verde, as superfícies escondidas de A em vermelho e as superfícies escondidas de B e C em azul. Em (b) o objeto A separado da cena. Em (c) a diferença entre A e a cena. Em (d) a interseção entre A e a cena.

Após a construção dos campos de distância das malhas de triângulos e de posse das funções implícitas das superfícies parametrizadas, utilizando as operações CSG apresentadas na Seção 2.2, é possível obter os campos de distância resultantes da diferença e da interseção entre um objeto e a cena. Para obter a diferença entre o objeto de interesse e a cena, basta construir um ADF utilizando como função de distância a operação de diferença entre o objeto e os outros N objetos que compõem a cena, ou seja, $dist(A - Cena) = \min(dist(A), -dist(i), -dist(i + 1), \dots, -dist(N))$. Já para obter a interseção

entre um objeto e a cena, a função de distância utilizada na construção do ADF é a união das interseções entre o objeto e cada um dos outros N objetos, ou seja, $dist(A \cap Cena) = (A \cap i) \cup (A \cap (i + 1)) \cup \dots \cup (A \cap N)$.

Essas operações podem ter um custo computacional elevado, se realizadas de maneira simplista na qual todos os objetos são testados contra todos. Para diminuir esse custo, uma BVH das AABBs dos objetos foi construída, de acordo com o exposto na Seção 2.3. Assim, os cálculos da diferença e da interseção podem ser realizados apenas com os objetos que podem influenciar na área de pintura de um objeto de interesse, e não com a cena completa.

Contudo, a abordagem descrita não trata bem o contato, pois não há interseção entre os sólidos. Para resolver esse problema, foi elaborado o conceito de tolerância de acesso: uma constante que indica um envelope dos objetos da cena, representando uma distância mínima de acesso físico para a realização da pintura, como pode ser observado na Figura 3.10. Utilizando essa tolerância na operação de interseção entre dois objetos, ela passa a ser dada por $dist(A \cap B) = \min(dist(A) + tolerância, dist(B) + tolerância)$.

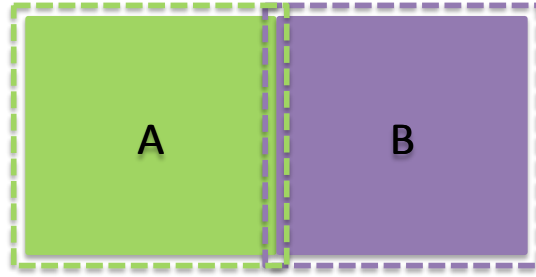


Figura 3.10: Cena com os objetos A e B, com a tolerância de acesso aplicada sobre os objetos, em tracejado, para o construção do ADF da interseção entre A e B.

Ao calcular a interseção com a tolerância de acesso, um erro é inserido no cálculo da área de interseção, visualizado na Figura 3.11; na medida em que ela aumenta de tamanho, devido ao aumento das superfícies escondidas da cena e também da superfície escondida do objeto. Assim a Equação 3-3 pode ser redefinida como:

$$A_i = A_{eo} + A_{ec} + erro \quad (3-5)$$

Aplicando em 3.4:

$$\begin{aligned} A_o + A_d - A_i &= (A_p + A_{eo}) + (A_p + A_{ec}) - (A_{eo} + A_{ec} + erro) \\ &= 2(A_p) - erro \\ \therefore A_p - \frac{erro}{2} &= \frac{A_o + A_d - A_i}{2} \end{aligned} \quad (3-6)$$

Dessa forma, o resultado obtido é menor que a área de pintura real, variando de acordo com o erro inserido pela tolerância.

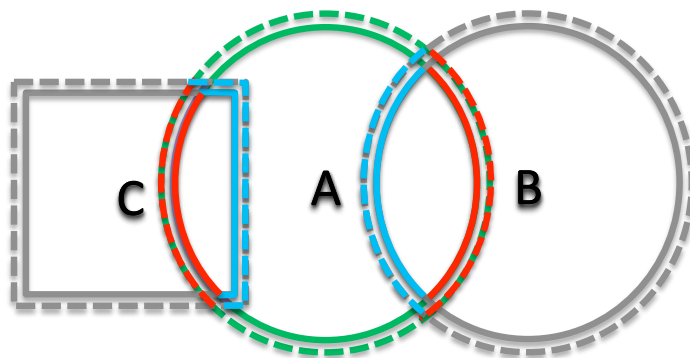


Figura 3.11: Tolerância aplicada na cena da Figura 3.9 para o cálculo da área de pintura de A, ressaltando o aumento da área da interseção, em tracejado, com a cena.

Para amenizar o erro inserido no cálculo da área, ocasionado pela utilização de uma tolerância na construção do ADF da interseção entre o objeto e a cena, o algoritmo de reconstrução da superfície de um ADF, apresentado por Frisken e Perry (21), foi alterado. No passo III, para posicionar o vértice de um triângulo gerado na superfície, o vértice sofre um deslocamento com o valor da distância reconstruída na direção do gradiente da superfície, que é um excelente indicativo da normal dessa superfície. A alteração consiste em executar esse deslocamento descontando a tolerância aplicada, retornando a superfície para sua posição anterior à aplicação da tolerância, sem o envelope, ilustrado em 2D na Figura 3.12.

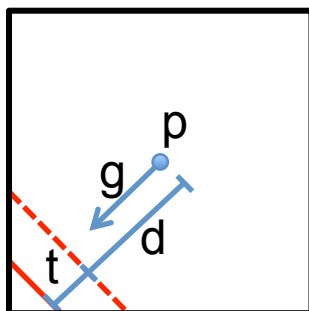


Figura 3.12: Vértice p sendo deslocado no sentido do gradiente g em direção da superfície, com um valor equivalente a distância reconstruída d , chegando na superfície com a tolerância de acesso aplicada (tracejada). Ao descontar a tolerância aplicada t , a superfície anterior a aplicação da tolerância é alcançada.

Uma vez que os campos de distância da diferença e da interseção, com tolerância de acesso, entre um objeto e a cena, são construídos, para que a área de pintura do objeto seja calculada, é necessário que as áreas das superfícies representadas por esses campos sejam computadas. Para isso, os triângulos que

compõem a superfície são reconstruídos utilizando o *SurfaceNets* estendido, apresentado na Seção 2.1.2. Entretanto reconstruir a superfície que o ADF representa para, em seguida, calcular a área de cada triângulo, pode demandar um alto custo de memória para armazenar a malha, na medida em que o ADF pode estar bem refinado na superfície, gerando uma quantidade considerável de triângulos.

Para evitar esse uso excessivo de memória, o algoritmo de reconstrução da superfície de um ADF foi mais uma vez modificado, para obter apenas a área da superfície reconstruída. Para isso, os passos II e III foram então condensados em um único passo, no qual, logo após um triângulo ser gerado, de acordo com os critério do passo II, seus vértices são movidos para a superfície e, depois de ter sua área calculada, é descartado. A área de cada triângulo gerado é acumulada, obtendo assim a área da superfície reconstruída.

Utilizando o algoritmo com as alterações indicadas anteriormente, a área da superfície representada por um ADF é calculada, sem a necessidade de armazenamento das malhas de triângulos resultantes da reconstrução da superfície do ADF. É ressaltado que no algoritmo modificado, uma parte do passo III não é executada: os vértices não são movidos no sentido da média de das posições dos seus vizinhos. Dessa forma, os triângulos gerados para o cálculo da área possuem uma qualidade inferior que a do algoritmo original. Contudo, como se está interessado apenas na área da superfície, obter triângulos mais próximos de triângulos equiláteros não é um fator decisivo para o cálculo.

Uma forma alternativa de calcular a área de pintura seria obter um ADF que representa a união entre todos os objetos de interesse da cena. Contudo, para calcular a área de muitos objetos, seria necessário um ADF grande e bem refinado, demandando assim uma quantidade excessiva de memória. Outro ponto negativo de computar a união de todos os objetos da cena, é o fato de se obter apenas uma área de pintura para todos os objetos, isto é, não é obtido a área de pintura de cada objeto individualmente, sendo que, obter essa área de maneira individual, é uma característica almejada, pois, dependendo das características dos objetos, eles são pintados com um tipo de tinta mais adequado.

4

Resultados e Discussão

Este capítulo apresenta os resultados obtidos através da utilização da abordagem proposta, baseada em campos de distância, para o cálculo da área de pintura de objetos em modelos CAD, tanto para cenas sintéticas controladas quanto para modelos reais. Além disso, é feita uma discussão sobre a eficiência do método proposto, bem como da redução da área de pintura obtida nos modelos reais, quando comparado ao método simplista que não trata as superfícies escondidas. Para a realização dos testes, a profundidade máxima do ADF foi fixado em 8, o erro tolerado para a reconstrução da distância em 10^{-4} e a tolerância de acesso fixada em 5×10^{-3} .

4.1

Modelos Controlados

Primeiramente, para validar o método proposto, modelos controlados foram elaborados, buscando simular os principais casos encontrados nos modelos CAD reais, como a interseção entre equipamentos, o contato de equipamentos sobrepostos e o contato entre as tubulações. Nesse sentido, três modelos foram produzidos: um para cada um dos casos comumente encontrados, sendo o primeiro um conjunto de esferas em interseção, o segundo um conjunto de caixas em contato e o terceiro um conjunto de cilindros também em contato.

4.1.1

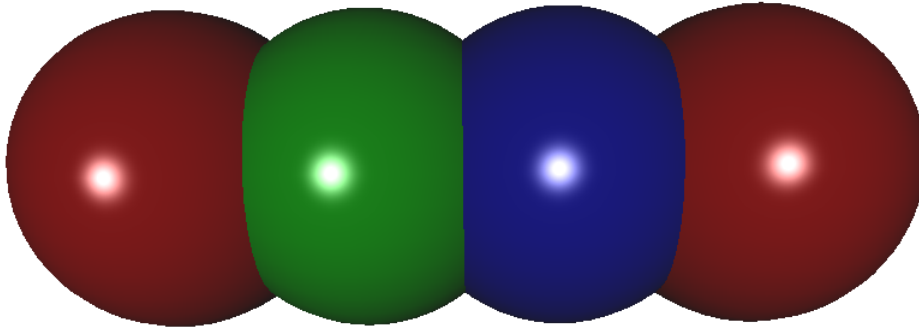
Esferas em Interseção

O primeiro modelo controlado desenvolvido foi um conjunto de esferas em interseção, como pode ser observado na Figura 4.1, visando simular os casos reais de interpenetração entre os objetos dos modelos CAD. Na cena elaborada, existem quatro esferas com interpenetração, todas com raio igual a 0,5. É possível notar a presença de superfícies escondidas, com a forma de calotas esféricas, as quais possuem raio igual a 0,5 e altura igual a 0,2.

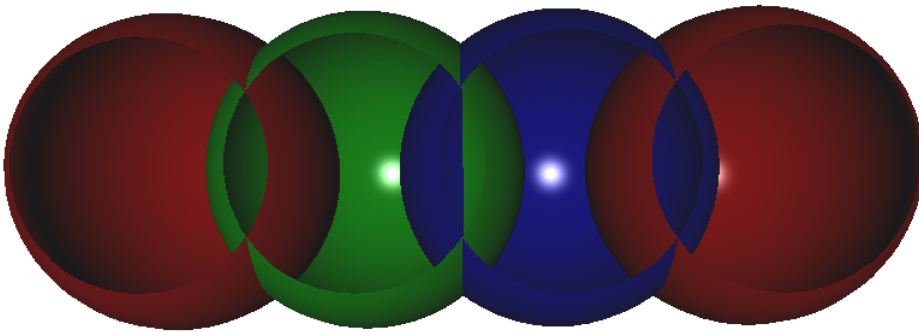
Na cena existem seis calotas esféricas encobertas: uma na primeira esfera vermelha, resultante da interseção com a esfera verde, duas na esfera verde, em interseção com a primeira esfera vermelha e com a esfera azul, duas na esfera azul, em interseção com a segunda esfera vermelha e com a esfera verde,

e uma na segunda esfera vermelha, em interseção com a esfera azul. A área de pintura dessa cena elaborada pode ser obtida, de maneira analítica, pela subtração da área das calotas esféricas da área total das esferas. Sendo a área da esfera dada por $4\pi r^2$, onde r é o raio da esfera, e a área da calota esférica dada por $2\pi rh$, onde h a altura da calota, a área de pintura da cena pode ser dada por:

$$A_p = 4(4\pi r^2) - 6(2\pi rh) \quad (4-1)$$



(a)



(b)

Figura 4.1: Modelo com quatro esferas de raio igual a 0,5, apresentando calotas esféricas, de altura 0,2, em interpenetração. Em (a), a cena pode ser observada de um ponto de vista externo. Em (b), o interior da cena pode ser visualizado.

Aplicando os valores de raio das esferas e altura das calotas, com $\pi = 3,1416$, é obtido uma área de pintura igual a 8,7964. Ao utilizar a abordagem de cálculo de pintura baseada em campos de distância, computando as diferenças e as interseções entre cada objeto e a cena, observadas na Figura 4.2, uma estimativa de área de pintura igual a 8,72237 foi obtida. Assim, a abordagem proposta demonstra tratar a interpenetração entre objetos de uma maneira satisfatória, na medida em que a estimativa obtida é apenas 0,84% menor que a área de pintura da cena.

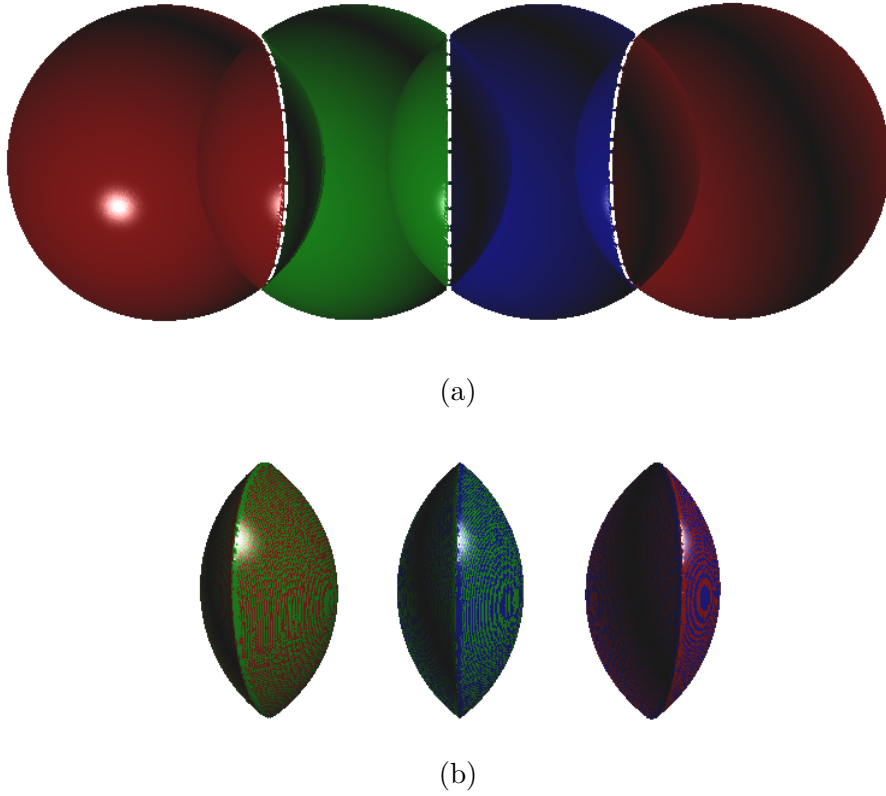


Figura 4.2: Em (a) é possível observar as diferenças entre cada objeto e a cena. Em (b), as interseções, entre cada objeto e a cena, são visualizadas.

4.1.2

Caixas em Contato

Com o objetivo de simular os casos reais de contato entre equipamentos e estruturas, como caixas em cima do piso e equipamentos sobrepostos ou lado a lado, foi elaborado um modelo controlado, com quatro caixas em contato, ilustrado na Figura 4.3. Na cena construída, quatro caixas em contato, todas com lado 0,5, produzem uma série de contatos entre as suas faces.

Nessa cena, é possível notar a existência de seis faces em contato: duas entre a primeira caixa vermelha e a caixa verde, uma de cada, duas entre a caixa verde e a azul, uma de cada, e duas entre a caixa azul e a segunda caixa vermelha, uma de cada também. A área de pintura pode ser computada analiticamente pela subtração da área das faces em contato da área total das caixas. A área da caixa é dado por $6l^2$, e a área de cada face em contato é dada por l^2 , onde l é o lado da caixa. Assim, a área de pintura dessa cena pode ser dada por:

$$A_p = 4(6l^2) - 6(l^2)$$

$$\therefore A_p = 18l^2 \quad (4-2)$$



(a)



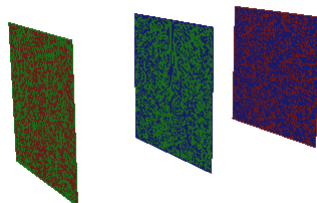
(b)

Figura 4.3: Modelo com quatro caixas de lado 0,5, no qual as faces das caixas estão em contato. Em (a), o exterior da cena pode ser observado e, em (b), o seu interior.

Substituindo o valor do lado da caixa, uma área de pintura igual a 4,5 é obtida. Utilizando a abordagem proposta de cálculo de pintura, através da construção das diferenças e das interseções entre cada objeto com a cena, visualizadas na Figura 4.4, foi obtida uma área de pintura igual a 4,47272, um resultado apenas 0,6% menor que a área de pintura computada analiticamente. Dessa maneira, a abordagem proposta se mostra apta para o tratamento de contato entre os equipamentos e estruturas dos modelos 3D.



(a)



(b)

Figura 4.4: Em (a), as diferenças entre cada caixa e a cena são visualizadas, e, em (b), as interseções.

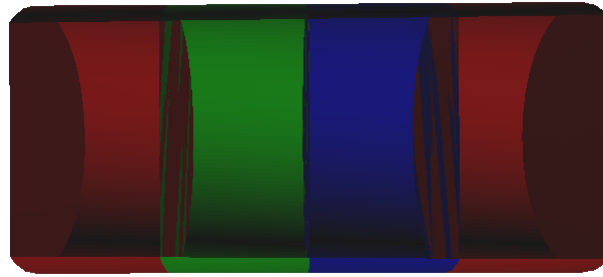
4.1.3

Cilindros em Contato

Por fim, foi gerado um último modelo para simular um caso bastante recorrente: os contatos que são produzidos durante a modelagem de tubulações. Para isso, cilindros de altura e raio 0,5, visualizados na Figura 4.5, foram colocados lado a lado, deixando suas tampas em contato, assim como observado nos modelos CAD.



(a)



(b)

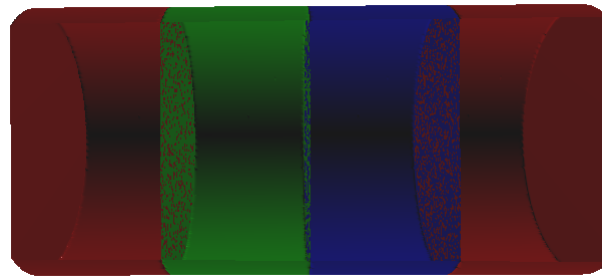
Figura 4.5: Modelo com quatro cilindros de raio e altura iguais 0,5, apresentando contato entre as tampas dos cilindros. Em (a), o exterior da cena é visualizado, enquanto em (b), é possível observar seu interior.

Nessa última cena controlada, é observado a presença de quatro cilindros, com seis tampas em contato, assim como no modelo das caixas, duas entre o primeiro cilindro vermelho e o cilindro verde, duas entre o cilindro verde e o azul e duas entre o cilindro azul e o segundo cilindro vermelho. Para calcular de maneira analítica a área de pintura desse modelo, basta subtrair da área total dos cilindros, a área das tampas em contato. A área do cilindro é dada pela soma entre a área da lateral, $2\pi rh$, e duas vezes a área da tampa, πr^2 , onde r é o raio do cilindro e h a sua altura. Dessa forma, a área de pintura da cena é dada por:

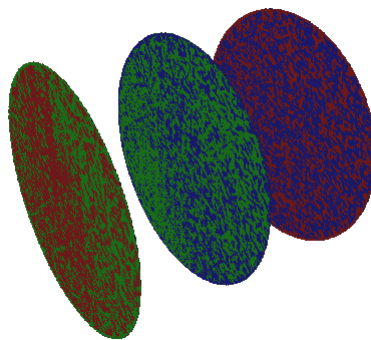
$$A_p = 4(2\pi r^2 + 2\pi rh) - 6(\pi r^2)$$

$$\therefore A_p = 8\pi rh + 2\pi r^2 \quad (4-3)$$

Ao substituir os valores de raio e altura dos cilindros, utilizando $\pi = 3,1416$, é obtido uma área de pintura igual a 7,85400. Aplicando a abordagem desenvolvida, e, obtendo as diferenças e interseções ilustradas na Figura 4.6, uma área de pintura igual a 7,81047, uma área de pintura apenas 0,55% menor do que a área calculada de forma analítica. Assim, mais uma vez, a abordagem proposta obtém um resultado satisfatório no tratamento de contato entre os objetos dos modelos 3D.



(a)



(b)

Figura 4.6: Em (a), as diferenças entre cada objeto e a cena são notadas, já em (b), é possível observar as interseções entre cada objeto e a cena.

4.2

Modelos Reais

Com a abordagem validada com cenas controladas, testes com modelos reais foram então realizados, a fim de estimar em quanto a abordagem pode reduzir a área de pintura, se comparada com a abordagem simplista de calcular a área superficial de cada objeto. O primeiro teste com modelos reais foi o cálculo da área de pintura de uma válvula, observada na Figura 4.7(a), composta por 23 objetos diferentes. Utilizando a abordagem simplista, foi obtido uma área de pintura igual a $0,3301\text{m}^2$, enquanto a abordagem proposta obteve uma estimativa igual $0,2423\text{m}^2$, ou seja, uma redução de 26,6%.

Em um segundo teste, também com uma válvula, composta de 12 objetos, ilustrada na Figura 4.7(b), foi obtida uma redução de 33,8%, com a área

calculada de maneira simplista igual a $2,3870\text{m}^2$ e a área estimada pela abordagem sendo igual a $1,5796\text{m}^2$. Em seguida a abordagem foi testada em um vaso, visualizado na Figura 4.8(a), composto de 73 objetos, obtendo uma redução de 38%, com o cálculo simplista resultando em uma área de $190,3764\text{m}^2$ e a estimativa com abordagem proposta resultando em $117,9540\text{m}^2$.

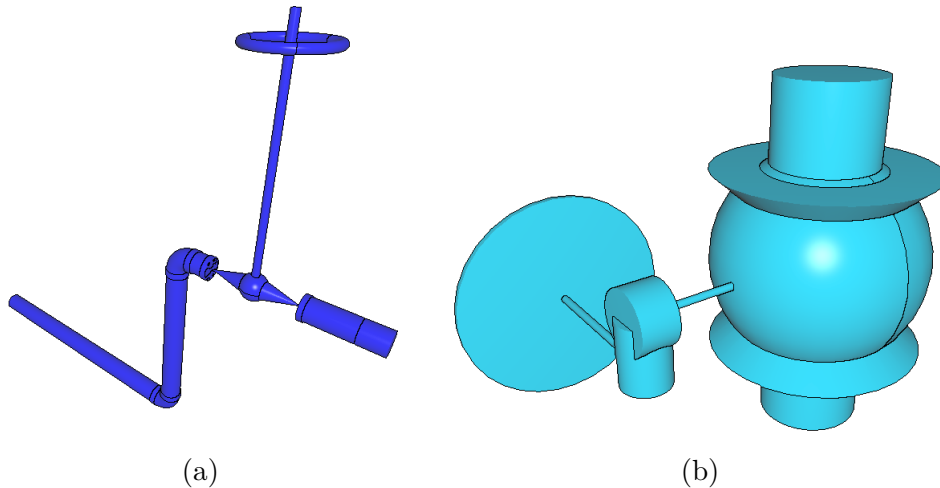


Figura 4.7: Em (a), uma válvula de um modelo real, composta por 23 objetos distintos, com uma área superficial de $0,3301\text{m}^2$. Em (b), uma válvula globo real, com 12 objetos diferentes e uma área superficial de $2,3870\text{m}^2$.

Após a realização desses testes em modelos com poucos elementos, foi realizado um teste com uma parte de uma unidade de compressão, composta de 373 objetos, observada na Figura 4.8(b). Nesse modelo foi observado uma redução de 17,5%, na medida em que a estimativa obtida com a abordagem proposta foi de $32,6232\text{m}^2$, enquanto a área calculada simplificada foi de $39,5700\text{m}^2$.

Por fim, testes com modelos maiores foram realizados. Nesses testes, uma região do modelo é isolada, e então a área dessa região é computada. Essas regiões, definidas durante o processo de manutenção das plataformas, indicam as zonas que serão pintadas durante a manutenção. Primeiramente foi realizado um teste em uma tubulação de uma zona, contendo 725 objetos. Utilizando a abordagem proposta, a estimativa de área de pintura foi de $19,3030\text{m}^2$, enquanto o cálculo simplista obtém uma área de $29,2283\text{m}^2$, resultando em uma redução de 33,9%.

Um segundo teste foi realizado, utilizando uma zona com tubulações e estruturas, somando 1.582 objetos. Nesse teste, a área calculada de maneira simplista foi de $396,6753\text{m}^2$ enquanto utilizando a abordagem proposta foi de $365,2450\text{m}^2$, 7,9% menor. A redução nessa zona é pequena pois as suas geometrias estão bem espaçadas, não havendo muitas interseções e contatos. Um terceiro teste, utilizando uma zona com 4.450 objetos, foi realizado. O

cálculo simplista dessa zona obtém uma área de $7.132,6500\text{m}^2$. Já abordagem proposta obteve uma estimativa de $6.551,4900\text{m}^2$, ou seja, um resultado 8,1% menor.

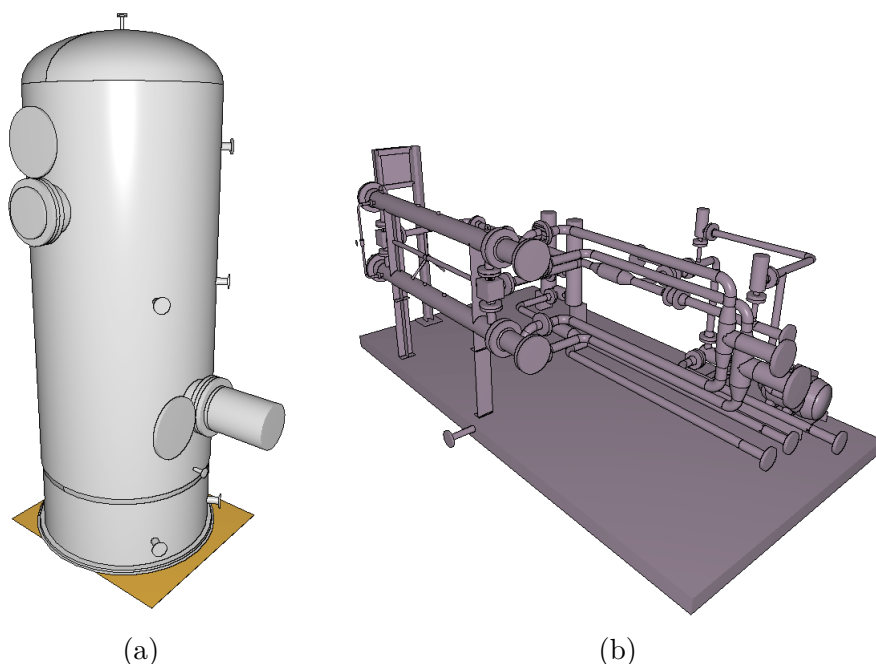


Figura 4.8: Em (a), um vaso, composto por 73 objetos distintos, com uma área superficial de $190,3764\text{m}^2$. Em (b), uma parte de uma unidade de compressão, com 373 objetos e uma área superficial de $39,5700\text{m}^2$.

4.2.1

Desempenho e Memória

Ao realizar os testes com os modelos reais, em uma máquina com 32GB de memória RAM e processador Intel(R) Core(TM) i7-4810MQ, foi percebido que a quantidade de memória necessária é influenciada pelo número de malhas, isto é, objetos que não são representados por superfícies parametrizadas, observado na Tabela 4.1, na medida em que os ADFs dessas malhas precisam ser construídos antes do cálculo da pintura, ou seja, quanto maior a quantidade de malhas, a memória consumida tende a aumentar.

Ainda na Tabela 4.1, foi notado que o tempo utilizado para o cálculo da área de pintura depende da quantidade de objetos total do modelo, pois todos os objetos tem suas diferenças e interseções com a cena calculadas, isto é, quanto maior o número de objetos no modelo, maior tende ser o tempo gasto na estimativa da área de pintura.

Contudo, o modelo 2 (Figura 4.7(b)), apesar de possuir menos objetos do que o modelo 1 (Figura 4.7(a)), obteve um tempo de execução maior. Isso se deve ao fato de que, no modelo 2, os objetos estão muito próximos, dessa

forma, a construção dos ADFs da diferença e da interseção de cada objeto com a cena é mais custosa, se comparada com o modelo 1, no qual os objetos estão mais espaçados.

Tabela 4.1: Consumo de memória e desempenho obtidos na aplicação da abordagem proposta em modelos reais.

Modelo	Nº Objetos	Nº Param.	Nº Malhas	Memória[MB]	Tempo[seg.]
1	23	22	1	94,37	16,48
2	12	10	2	69,62	18,12
3	73	35	38	689,87	119,50
4	373	314	59	697,45	216,25
5	725	427	298	1.986,56	577,42
6	1.582	435	1.147	11.448,32	1.656,38
7	4.450	844	3.606	24.094,72	5.655,53

Com o objetivo de observar o comportamento da abordagem com uma profundidade máxima do ADF diferente, um teste com o modelo 7 foi realizado, diminuindo a profundidade de 8 para 7. Nesse teste, a área de pintura estimada pela abordagem proposta foi de 6.578,5800, 0,4% maior que a área obtida com a profundidade igual a 8, sendo que, para isso, levou 1.371,27 segundos, uma redução de 75,7% se comparada com tempo anterior. Além disso, a quantidade de memória utilizada foi equivalente a 14.069,76 MB, 41,6% menor que a memória consumida no teste com profundidade 8.

4.3

Discussão

Analisando os resultados obtidos, é possível observar que a abordagem desenvolvida para estimar a área de pintura, obteve resultados satisfatórios, tanto nos modelos controlados quanto nos modelos reais. Nas cenas que simulam as principais ocorrências de superfícies encobertas, os resultados obtidos diferenciaram de, no máximo, 0,84%, demonstrando a capacidade da abordagem de estimar a área de pintura satisfatoriamente. É importante destacar que a área de pintura estimada pela abordagem proposta tende a ser menor que área real, devido ao erro inserido pela tolerância de acesso.

Já nos modelos reais, foi possível obter uma redução de até 38%, em relação à abordagem simplista de calcular a área superficial de cada objeto. Essa redução, mesmo nos casos em que as geometrias estão mais esparsas, é significativa, pois, além de ser uma atividade com um custo elevado, a pintura

de 1m^2 possui um custo estimado de R\$ 219,00, é uma tarefa recorrente na manutenção de plataformas petrolíferas.

Assim, a longo prazo, a economia de recursos pode ser considerável, caso a área de pintura seja estimada de acordo com a abordagem desenvolvida. É ressaltado que, apesar da redução na zona com 4.450 objetos ter sido de 8,1%, foi obtida uma redução de $581,1600\text{m}^2$ na área de pintura, possibilitando uma economia expressiva na manutenção da pintura anticorrosiva dessa zona, estimada em R\$ 127.274,04.

Outro ponto importante é consumo de memória, quando o modelo possui uma grande quantidade de malhas. Nesses modelos, a quantidade de memória utilizada para a estimativa da área de pintura ainda foi elevada, devido à construção prévia de todos os ADFs que representam essas malhas.

Contudo foi observado que, com a redução de um nível na profundidade máxima da ADF, a quantidade de memória exigida, assim como o tempo gasto, foram reduzidos consideravelmente. Essa redução acontece pois, com um ADF de menor profundidade, a quantidade de células geradas é bem menor, na medida em que cada célula do ADF se subdivide em 8 novas células ao descer um nível de profundidade.

Assim o consumo de memória e o tempo demandado para a construção dos ADFs das malhas, e das diferenças e interseções com a cena, são diminuídos. É importante ressaltar que a redução de um nível na profundidade máxima da ADF tende a produzir resultados menos precisos, pois células folha que possuem um erro maior que o erro tolerado não são subdivididas.

Este trabalho apresentou uma abordagem para estimar a área de pintura dos diversos elementos encontrados em modelos CAD, sem levar em consideração a área de superfícies escondidas, seja pelo contato ou pela interpenetração entre objetos. A abordagem proposta explorou uma relação percebida entre a área superficial do objeto e as áreas da diferença e da interseção do objeto com a cena.

Campos de distância adaptativo foram utilizados, para obter as áreas das diferença e da interseção do objeto com a cena, pois são muito adequados para a aplicação de operações CSG, devido ao fato de proporcionar, de maneira eficiente, a menor distância entre um ponto e a superfície representada. Além disso, árvores, como a *k-d tree* e a BVH de AABB, foram empregadas, no sentido de acelerar a construção do ADF das malhas, no caso da *k-d tree*, e também para acelerar o cálculo da diferença e da interseção entre um objeto e a cena. É ressaltado que para a construção do ADF de uma malha, a mesma deve ser fechada, o que representa uma limitação da abordagem proposta.

Para tratar o problema do contato, o conceito de tolerância de acesso foi introduzido. Apesar dessa tolerância inserir um erro na estimativa de área, esse erro foi amenizado, através de modificações no algoritmo de reconstrução da superfície de um ADF. Além dessa modificação, o algoritmo também foi adaptado para calcular a área da superfície representada, sem o consumo excessivo de memória.

Os resultados obtidos com os modelos controlados demonstram a capacidade da abordagem de estimar, satisfatoriamente, a área de pintura de uma cena, através do cálculo da área de pintura de cada elemento. Nesses modelos, a abordagem estimou a área de pintura com uma precisão que variou de 99,16% até 99,45%. Ao estimar a área de pintura de modelos reais com a abordagem baseada em campos de distância, foi obtido uma redução de até 38% da área de pintura de uma cena, quando comparada com a abordagem simplista de computar a área superficial de cada objeto.

Assim, de maneira geral, utilizar a abordagem proposta para a estimativa de área de pintura em modelos CAD pode levar a uma economia considerável de recursos, na medida em que a manutenção da pintura anticorrosiva é uma

atividade constante, principalmente em plataformas petrolíferas, pois estão sujeitas às fortes intempéries do alto mar.

É ressaltado que os ADFs construídos para representar as malhas de triângulos são baseados em uma caixa alinhada com os eixos. Assim, geometrias que não são alinhadas com os eixos produzem uma série de células que poderiam ser evitadas com a utilização de outros tipos de caixa envolvente. Dessa forma, uma proposta de trabalho futuro seria a utilização de ADFs baseadas em caixas orientadas com os eixos do objeto.

Um outro ponto importante é a necessidade da construção prévia de todos os ADFs das malhas de triângulos, para obter a diferença e a interseção entre um objeto e a cena, o que pode provocar um consumo de memória elevado. Assim, uma segunda proposta de trabalho futuro seria desenvolver uma abordagem que construísse os ADFs sob demanda, e destruísse os ADFs apenas quando, garantidamente, não forem mais necessários para o cálculo dos objetos que ainda não tiveram sua área calculada.

Uma terceira proposta de trabalho futuro, seria definir a profundidade máxima da ADF de maneira adaptativa, ou seja, o ADF é refinado até que a diferença entre as áreas das superfícies reconstruídas nos níveis N e $N - 1$ seja menor que uma tolerância previamente estabelecida. Assim, o nível de refinamento de um ADF é definido pela presença ou ausência de detalhes relevantes na superfície que ele representa, seja uma malha, ou a diferença e a interseção de um objeto com a cena.

É importante salientar que a abordagem proposta pode ser utilizada com outros enfoques, como a validação do projeto conceitual antes da construção. Nesse exemplo, a abordagem desenvolvida pode ser utilizada para identificar regiões de difícil acesso para a pintura, evitando assim a construção de uma plataforma, por exemplo, que possua áreas cuja manutenção com tinta anticorrosiva não seja realizada, gerando assim um desgaste maior da planta.

Além disso, o cálculo da área de pintura de um modelo pode ser utilizado não só para estimar a quantidade de tinta que é necessária para realizar a manutenção de instalações industriais, podendo ser utilizado também para estimar a quantidade de material necessário para imprimir um modelo em uma impressora 3D. Nesse caso, a economia de material utilizado pode impactar diretamente no custo da impressão, dependendo do tipo e qualidade do material.

Referências bibliográficas

- [1] BÉZIER, PIERRE E. **Example of an existing system in the motor industry: the UNISURF system**. In: PROCEEDINGS OF THE ROYAL SOCIETY OF LONDON A: MATHEMATICAL, PHYSICAL AND ENGINEERING SCIENCES, volumen 321, p. 207–218. The Royal Society, 1971.
- [2] EASTMAN, CHARLES M AND EASTMAN, CHUCK AND TEICHOLZ, PAUL AND SACKS, RAFAEL. **BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors**. John Wiley & Sons, 2011.
- [3] JONES, MARK W AND BAERENTZEN, J ANDREAS AND SRAMEK, MILOS. **3D distance fields: A survey of techniques and applications**. IEEE Transactions on visualization and Computer Graphics, 12(4):581–599, 2006.
- [4] GHALI, SHERIF. **Constructive solid geometry**. Introduction to Geometric Computing, p. 277–283, 2008.
- [5] XU, HONGYI AND BARBIČ, JERNEJ. **Signed distance fields for polygon soup meshes**. In: PROCEEDINGS OF GRAPHICS INTERFACE 2014, p. 35–41. Canadian Information Processing Society, 2014.
- [6] ROSENFELD, AZRIEL AND PFALTZ, JOHN L. **Sequential operations in digital picture processing**. Journal of the ACM, 13(4):471–494, 1966.
- [7] KAUFMAN, ARIE AND COHEN, DANIEL AND YAGEL, RONI. **Volume graphics**. Computer, 26(7):51–64, 1993.
- [8] GUENDELMAN, ERAN AND BRIDSON, ROBERT AND FEDKIW, RONALD. **Nonconvex rigid bodies with stacking**. In: ACM TRANSACTIONS ON GRAPHICS, volumen 22, p. 871–878. ACM, 2003.
- [9] ERLEBEN, KENNY. **Stable, robust, and versatile multibody dynamics animation**. Ph. D. Thesis, University of Copenhagen, Copenhagen, 2004.
- [10] HIROTA, GENTARO AND FISHER, SUSAN AND OTHERS. **An improved finite-element contact model for anatomical simulations**. The Visual Computer, 19(5):291–309, 2003.

- [11] BRIDSON, ROBERT AND MARINO, SEBASTIAN AND FEDKIW, RONALD. **Simulation of clothing with folds and wrinkles**. In: PROCEEDINGS OF THE 2003 ACM SIGGRAPH/EUROGRAPHICS SYMPOSIUM ON COMPUTER ANIMATION, p. 28–36. Eurographics Association, 2003.
- [12] SERRA, JEAN. **Image analysis and mathematical morphology, v. 1**. Academic press, 1982.
- [13] JONES, MARK W. **Facial reconstruction using volumetric data**. In: VMV, p. 135–150, 2001.
- [14] BREEN, DAVID E AND MAUCH, SEAN AND WHITAKER, ROSS T. **3D scan conversion of CSG models into distance volumes**. In: PROCEEDINGS OF THE 1998 IEEE SYMPOSIUM ON VOLUME VISUALIZATION, p. 7–14. ACM, 1998.
- [15] BÆRENTZEN, ANDREAS AND CHRISTENSEN, NIELS JØRGEN. **A technique for volumetric CSG based on morphology**. In: PROCEEDINGS OF INTERNATIONAL WORKSHOP ON VOLUME GRAPHICS, p. 117–130, 2001.
- [16] TERAN, JOSEPH AND MOLINO, NEIL AND FEDKIW, RONALD AND BRIDSON, ROBERT. **Adaptive physics based tetrahedral mesh generation using level sets**. Engineering with Computers, 21(1):2–18, 2005.
- [17] PERSSON, PER-OLOF. **Mesh generation for implicit geometries**. PhD thesis, Massachusetts Institute of Technology, 2005.
- [18] FRISKEN, SARAH F AND PERRY, RONALD N AND ROCKWOOD, ALYN P AND JONES, THOUIS R. **Adaptively sampled distance fields: A general representation of shape for computer graphics**. In: PROCEEDINGS OF THE 27TH ANNUAL CONFERENCE ON COMPUTER GRAPHICS AND INTERACTIVE TECHNIQUES, p. 249–254. ACM Press/Addison-Wesley Publishing Co., 2000.
- [19] SAMET, HANAN. **The design and analysis of spatial data structures**, volumen 199. Addison-Wesley Reading, MA, 1990.
- [20] DE FIGUEIREDO, LUIZ HENRIQUE AND VELHO, LUIZ AND DE OLIVEIRA, JOÃO BATISTA. **Revisiting adaptively sampled distance fields**. In: COMPUTER GRAPHICS AND IMAGE PROCESSING, 2001 PROCEEDINGS OF XIV BRAZILIAN SYMPOSIUM ON, p. 377. IEEE, 2001.

- [21] PERRY, RONALD N AND FRISKEN, SARAH F. **Kizamu: A system for sculpting digital characters**. In: PROCEEDINGS OF THE 28TH ANNUAL CONFERENCE ON COMPUTER GRAPHICS AND INTERACTIVE TECHNIQUES, p. 47–56. ACM, 2001.
- [22] GIBSON, S. **Using distance maps for accurate surface reconstruction in sampled volumes**. In: 1998 SYMPOSIUM ON VOLUME VISUALIZATION, p. 23–30, 1998.
- [23] BLOOMENTHAL, JULES. **Polygonization of implicit surfaces**. Computer Aided Geometric Design, 5(4):341–355, 1988.
- [24] LORENSEN, WILLIAM E AND CLINE, HARVEY E. **Marching cubes: A high resolution 3D surface construction algorithm**. In: ACM SIGGRAPH COMPUTER GRAPHICS, volumen 21, p. 163–169. ACM, 1987.
- [25] KARRON, DANIEL B AND COX, JAMES AND MISHRA, BHUBANESWAR. **New findings from the SpiderWeb algorithm: toward a digital morse theory**. In: VISUALIZATION IN BIOMEDICAL COMPUTING 1994, p. 643–657. International Society for Optics and Photonics, 1994.
- [26] HOFFMANN, CHRISTOPH MARTIN. **Geometric and solid modeling**. 1989.
- [27] WANG, SIDNEY W AND KAUFMAN, ARIE E. **Volume-sampled 3D modeling**. IEEE Computer Graphics and Applications, 14(5):26–32, 1994.
- [28] KAUFMAN, ARIE AND COHEN, DANIEL AND YAGEL, RONI. **Volume graphics**. Computer, 26(7):51–64, 1993.
- [29] FRISKEN, SARAH F AND PERRY, RONALD N. **Designing with distance fields**. In: ACM SIGGRAPH 2006 COURSES, p. 60–66. ACM, 2006.
- [30] ERICSON, CHRISTER. **Real-time collision detection**. CRC Press, 2004.
- [31] BERGEN, GINO VAN DEN. **Efficient collision detection of complex deformable models using AABB trees**. Journal of Graphics Tools, 2(4):1–13, 1997.
- [32] PAYNE, BRADLEY A AND TOGA, ARTHUR W. **Distance field manipulation of surface models**. IEEE Computer Graphics and Applications, 12(1):65–71, 1992.
- [33] BENTLEY, JON LOUIS. **Multidimensional binary search trees used for associative searching**. Communications of the ACM, 18(9):509–517, 1975.

- [34] FRIEDMAN, JEROME H AND BENTLEY, JON LOUIS AND FINKEL, RAPHAEL ARI. **An algorithm for finding best matches in logarithmic expected time.** ACM Transactions on Mathematical Software, 3(3):209–226, 1977.
- [35] AANÆS, HENRIK AND BÆRENTZEN, JAKOB ANDREAS. **Pseudo-Normals for Signed Distance Computation.** In: VISION, MODELING, AND VISUALIZATION 2003, MUNICH, GERMANY, 2003.
- [36] BÆRENTZEN, J. A.; AANAES, H.. **Signed distance computation using the angle weighted pseudonormal.** IEEE Transactions on Visualization and Computer Graphics, 11(3):243–253, 2005.
- [37] THÜRRNER, GRIT AND WÜTHRICH, CHARLES A. **Computing vertex normals from polygonal facets.** Journal of Graphics Tools, 3(1):43–46, 1998.