

5 Implementação e resultados

Nesta seção, apresentamos alguns detalhes de implementação das diferentes técnicas investigadas e comparamos os resultados obtidos em termos de facilidade de implementação, qualidade visual obtida e eficiência computacional, sempre levando em conta a aplicação a modelos CAD.

Os testes foram realizados em uma máquina com processador Intel Core 2 Duo E6600 2.4GHz e placa gráfica Nvidia GeForce 8800 GTS. A visualização foi realizada em dois modelos CAD de plataformas de petróleo com complexidade geométrica mediana (250 mil e 480 mil triângulos), em resolução 800×600, com mapas de sombra de 2048×2048. A geração dos mapas foi realizada em FBOs (*Framebuffer Objects*), necessitando de apenas uma passada por fonte de luz, exceto quando especificado. A renderização final foi realizada sempre com uma passada para a iluminação ambiente e a marcação do *z-buffer*, mais uma passada para a aplicação do mapa de sombras de cada fonte de luz, em todos os métodos implementados.

O grafo de cena com que os algoritmos foram integrados oferecia um mecanismo eficiente de descarte em relação ao volume de visão (*frustum culling*), que pôde ser utilizado em conjunto com as técnicas de reparametrização e particionamento. No entanto, não havia mecanismos de nível de detalhe (LOD, *level-of-detail*) e descarte por visibilidade (*visibility culling*). Essas técnicas de aceleração seriam úteis porque, na visualização de modelos CAD, muitas vezes toda a cena é vista pela câmera e iluminada pela fonte de luz, tornando o descarte em relação ao volume de visão ineficaz tanto na geração dos mapas de sombra quanto na renderização final. Assim, para compensar a ausência desses mecanismos e diminuir o gargalo de geometria da aplicação, as medidas de eficiência foram tomadas em pontos-de-vista restritos, onde o descarte em relação ao volume de visão garantia a renderização de apenas 20 a 30% da geometria da cena. Isso permitiu avaliar melhor o custo real dos algoritmos em si, cujo desempenho é usualmente limitado por rasterização nas passadas finais.

A Tabela 2 mostra a velocidade conseguida na renderização sem sombras e com mapeamento de sombras tradicional, podendo ser usada como referência no restante deste capítulo. Note que sombras dinâmicas causam uma grande queda de desempenho devido à atualização quadro-a-quadro dos mapas de sombra.

Tabela 2 – Medidas de velocidade de referência para a visualização (em FPS)

Map. Tradicional	Cena completa	Ponto-de-vista restrito
Sem sombras	60	300
1 luz estática	37	230
2 luzes estáticas	25	160
1 luz dinâmica	22	45
2 luzes dinâmicas	13	25

5.1. Técnicas anti-chamuscamento

Na Seção 3.1, discutimos as principais técnicas utilizadas para evitar o efeito de chamuscamento no mapeamento de sombras (veja a Figura 4 e a Tabela 1). Relatamos aqui os resultados de testes realizados em modelos CAD reais.

Inicialmente, notamos que as técnicas de *bias* constante e de *second-depth* podem ser implementadas com muita facilidade. Basta-se utilizar, na geração do mapa, as funcionalidades de “*polygon offset*” e de “*frontface culling*” do hardware gráfico, respectivamente. Assim, esses métodos não requerem nenhum custo adicional. As técnicas de ponto médio (c,d da Tabela 1), por outro lado, requerem duas passadas de renderização para a obtenção da segunda superfície por meio de “descamação de profundidades” (*depth-peeling*). Esses métodos, portanto, têm implementação mais trabalhosa e reduzem a eficiência do algoritmo, como mostra a Tabela 3. Isso confirma as conclusões da Seção 3.1.1.

Comparando as técnicas de *bias* constante e de *second-depth*, ressaltamos na Seção 3.1.1 que a segunda requer uma cena composta apenas de sólidos fechados, o que não acontece sempre no caso de modelos CAD. Outra desvantagem do *second-depth* refere-se à qualidade das sombras, como mostra a Figura 28. Na região A, a parte superior de um pequeno degrau, aparecem artefatos de chamuscamento em virtude da proximidade entre a primeira e a segunda superfícies. Nas regiões B e C, por outro lado, ocorre o fenômeno de vazamento de luz devido à combinação da técnica com o filtro PCF. O efeito aparece devido

ao fato de o filtro considerar também pontos da superfície internos aos cilindros, cujo teste de sombra resulta em iluminação total.

Tabela 3 – Medidas de velocidade de métodos anti-chamuscamento (em FPS)

Tipo de Sombras	Bias constante e <i>second-depth</i>		Técnicas de ponto médio	
	Cena completa	Ponto-de-vista restrito	Cena completa	Ponto-de-vista restrito
1 luz dinâmica	22	45	16	27
2 luzes dinâmicas	13	25	9	14

Na verdade, a melhor maneira de se evitar o chamuscamento é aumentar o aproveitamento da resolução do mapa de sombras, seja com técnicas de reparametrização e particionamento ou outras. Como esses métodos requerem a atualização do mapa a cada quadro mesmo com luzes estáticas e podem exigir múltiplas passadas no processo de geração, as técnicas de ponto-médio tornam-se muito custosas em modelos CAD com elevada complexidade geométrica. Em nosso sistema, a utilização dessas técnicas permitia o desempenho apenas a taxas interativas. Buscando-se a visualização em tempo real e descartando-se a técnica de *second-depth* devido aos artefatos da Figura 28 e à exigência de sólidos fechados, optamos por utilizar simplesmente um *bias* constante na geração dos mapas.

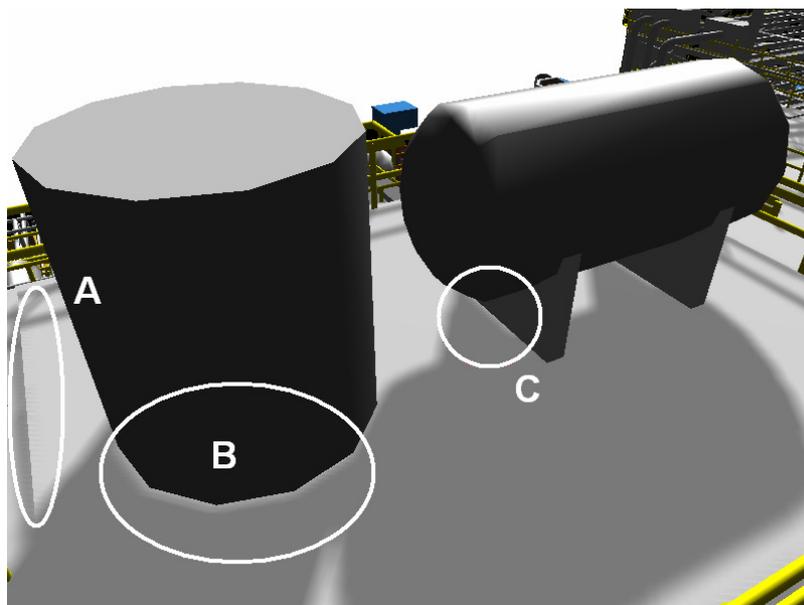


Figura 28 – Efeitos de chamuscamento (A) e vazamento de luz (B,C) com *second-depth*

5.2. Técnicas de filtragem

Vimos, na Seção 3.2, duas técnicas de filtragem para o tratamento do serrilhamento associado aos mapas de sombras: os mapas de variância (VSMs) e o filtro PCF. Discutimos aqui os resultados dos testes realizados com ambas técnicas.

A implementação de VSM seguiu a descrição de Donnelly & Lauritzen [18], utilizando para os mapas de sombras texturas de 4 componentes de ponto flutuante em 32 bits (não-filtráveis por hardware). Foi implementado um programa de fragmento para aplicar aos mapas um filtro gaussiano separável de tamanho até 5×5 , bem como uma função para a obtenção de amostras com filtragem bilinear, complementando a funcionalidade disponível em hardware. Infelizmente, como foi explicado na Seção 3.2 e ilustrado na Figura 10, a técnica resultou em artefatos sérios de vazamento de luz. Assim, para modelos CAD com grande complexidade em profundidade, sua utilização mostrou-se impraticável.

A filtragem PCF foi extensamente investigada. Em primeiro lugar, percebemos que placas gráficas Nvidia aplicam diretamente o PCF 2×2 em texturas de sombra com filtro ativado. Quando se desejam filtros com núcleos maiores, pode-se manter a opção ativa e implementar a amostragem adicional num programa de fragmento. Como o hardware realiza interpolação bilinear para os resultados dos testes de sombra, o efeito de faixas (*banding*) não aparece, sendo desnecessária a aplicação de *jitter* ou de um núcleo circular. Em placas ATI, por outro lado, essas técnicas são necessárias para suavizar corretamente as bordas das sombras. Recomendamos aqui um núcleo de tamanho 3×3 a 5×5 .

O plano tangente discutido nas Seções 3.1.2 e 3.2.2 foi utilizado para reduzir o efeito de chamuscamento, tomando-se amostras virtuais alinhadas com amostras do núcleo do filtro PCF. Quanto à amostragem adaptativa da Seção 3.2.2, constatamos que proporciona um ganho de eficiência de até 10%, com ganho maior quando o gargalo da aplicação está inteiramente no processo de rasterização. Os artefatos associados à amostragem adaptativa são minimizados, muitas vezes tornando-se imperceptíveis, após a reparametrização do mapa de sombras.

A Tabela 4 mostra a eficiência, em quadros por segundo, conseguida com os algoritmos de VSM e PCF. Como podemos ver, o tamanho dos núcleos utilizados não influenciou o desempenho em nenhum caso. Isso pode ser explicado por dois motivos: primeiro, em todas as passadas de geração de mapas de sombra, o gargalo encontra-se fortemente no processamento de geometria; segundo, a placa de vídeo utilizada nos testes possui elevado poder de rasterização de fragmentos, o que mantém a última passada de renderização rápida mesmo com núcleos maiores. Comparando as duas técnicas implementadas, o método VSM se destaca quando o mapa não precisa ser atualizado a todo quadro, já que a filtragem é realizada apenas uma vez. Com luzes dinâmicas ou técnicas de reparametrização, o ganho do VSM torna-se menor devido à aplicação repetida do filtro gaussiano às quatro componentes de mapas de grande resolução (no caso, 2048×2048).

Tabela 4 – Eficiência de técnicas de filtragem (em FPS, sempre ponto-de-vista restrito)

Tipo de Sombras (mapa simples)	PCF adaptativo		VSM	
	Hardware + PCF 3×3	Hardware + PCF 5×5	VSM + Gauss 3×3	VSM + Gauss 5×5
1 luz estática	84	84	194	194
2 luzes estáticas	56	56	116	116
1 luz dinâmica	43	43	46	46
2 luzes dinâmicas	22	21	25	25

5.3.

Técnicas de reparametrização e particionamento

Nesta seção, apresentamos detalhes de implementação e resultados de testes realizados com as técnicas de reparametrização e particionamento estudadas no Capítulo 4. A reparametrização foi baseada sempre no método TSM, com cálculos e manipulações inteiramente em duas dimensões. O parâmetro n' foi escolhido como n'_{LISPSM} , testado em suas versões original e generalizada.

Primeiramente, realizamos o ajuste dos planos *near* e *far* da câmera ao volume envolvente da cena completa (uma caixa alinhada ou AABB), de modo a maximizar o aproveitamento da resolução em profundidade do mapa de sombras. Em seguida, a caixa envolvente da cena e o volume de visão da câmera são transformados para o espaço projetado da fonte de luz, onde é calculada a interseção dos polígonos resultantes entre si e com o volume de visão da própria fonte. Definimos assim nossa região de interesse, que também é um polígono convexo.

O próximo passo é determinar o trapezóide associado à transformação perspectiva \mathbf{P} de reparametrização (Figura 17 da Seção 4.2). Assume-se que sua “linha central” passe pelos centros dos planos *near* e *far* da câmera. Suas bases são então retas perpendiculares a essa linha central, tocando o polígono de interesse. Finalmente, o centro de projeção \mathbf{c} é encontrado sobre a linha central, à distância n' da base menor do trapezóide.

A determinação do parâmetro n' segue a Equação (27) para n'_{LISPSM} ou, aplicando-se a forma generalizada, as Equações (42) e (44). Para garantir a estabilidade dos cálculos, impomos ainda um limite para a abertura dos trapezóides gerados pelo algoritmo. No caso de uma abertura excessiva, a distância n' é ligeiramente aumentada.

Uma vez determinado o trapezóide, a transformação perspectiva \mathbf{P} é obtida seguindo-se o procedimento descrito por Martin & Tan [23]. A idéia por eles apresentada para se manter os valores de z não transformados também é utilizada, de modo a se evitar a piora do efeito de chamuscamento. A única alteração feita ao restante do algoritmo de mapeamento de sombras tradicional é a multiplicação da matriz de \mathbf{P} pela matriz de projeção da fonte de luz antes de gerar o mapa.

A Figura 29 ilustra a diferença de resultado obtida com a utilização de n'_{LISPSM} generalizado em comparação com o parâmetro original. A luz e a câmera estão orientadas com um ângulo γ bastante próximo ao limite γ_{lim} , mas maior que ele, de forma que o algoritmo generalizado ainda não tenha começado a convergir para o mapeamento tradicional. Como se pode notar, a melhora de qualidade é bastante sutil, podendo ser bem percebida apenas quando as sombras são visualizadas a partir de um ponto muito próximo. A pequena diferença é esperada, já que o método realiza apenas uma pequena correção ao parâmetro n' . No entanto, como o custo extra da técnica é praticamente inexistente, sua aplicação ainda é recomendada.

Como o único custo associado a qualquer reparametrização perspectiva é o cálculo de \mathbf{P} em CPU, a técnica não afeta a eficiência do algoritmo de sombras. Essa afirmação é corroborada pela comparação entre a Tabela 2 e a coluna “ ZP_1 ” da Tabela 5, pela qual podemos perceber que pode haver ainda um possível ganho de desempenho com a reparametrização. Esse ganho decorre do descarte realizado na geração do mapa em relação ao campo de visão da fonte de luz ajustado ao

trapezóide (*frustum culling*). Mais uma vez, ressaltamos que nosso principal gargalo de desempenho está em processamento de geometria.

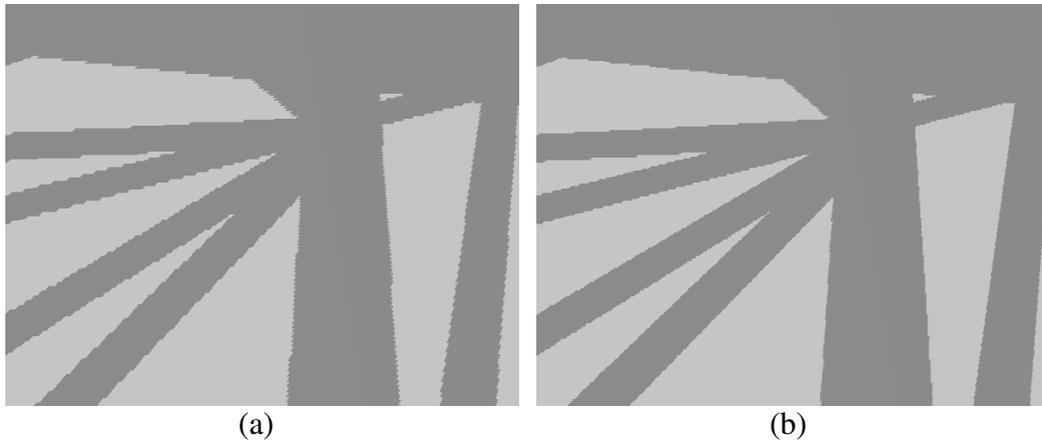


Figura 29 – Comparação entre as qualidades das sombras obtidas com a reparametrização do mapa utilizando n'_{LiSPSM} : (a) original e (b) generalizado.

Tabela 5 – Eficiência de técnicas de reparametrização e particionamento (em FPS)

Tipo de Sombras	Reparametrização sem particionamento (ZP_1)		Reparametrização e Particionamento em z (ZP_2)	
	Cena completa	Ponto-de-vista restrito	Cena completa	Ponto-de-vista restrito
1 luz dinâmica	21 a 22	44 a 88	16 a 20	31 a 66
2 luzes dinâmicas	13	30 a 49	9 a 12	22 a 38

Nos voltamos agora à técnica de particionamento em profundidade. Em nossa implementação, utilizamos duas partições definidas dinamicamente para evitar ao máximo o efeito de costura. Na Equação (55), utilizamos $\alpha_2 = 2$ para a segunda partição. A Tabela 5 mostra que a aplicação do particionamento reduz a eficiência do algoritmo, pois exige duas (ou mais) passadas de atualização do mapa.

As Figuras 30 a 32 ilustram o ganho de qualidade visual obtido com as técnicas de reparametrização e particionamento. Por sua eficiência, qualidade e relativa facilidade de implementação, a reparametrização do mapa de sombras é sempre recomendada. O ganho do particionamento em z, por outro lado, não é tão marcante (a não ser que se permitam costuras mais visíveis), mas ajuda a garantir a correta visualização de sombras de estruturas finas e de silhuetas complexas, principalmente em configurações com a luz e a câmera quase paralelas (note a diferença entre as imagens inferiores das Figuras 31 e 32). Além disso, pode-se conseguir uma melhora mais significativa alocando-se um mapa de sombras

completo para cada partição (em nossa implementação, as partições dividem a resolução de um único mapa). Assim, recomenda-se a utilização de duas partições com o esquema adaptativo proposto neste trabalho.

5.4. Sombras em movimento

Como vimos, as técnicas de reparametrização e particionamento do mapa de sombras alteram a amostragem de profundidades da cena. A determinação da transformação e das subdivisões utilizadas depende das posições da fonte de luz e da câmera. Assim, a qualidade das sombras pode sofrer variações quando a fonte ou a câmera se movimentam, causando um efeito de “*aliasing*” temporal conhecido como “sombras rastejantes” (*shadow crawling* ou *shadow swimming*): os artefatos de serrilhamento das bordas das sombras (e de chamuscamento, caso existam) se deslocam e mudam de tamanho de um quadro para outro, o que é desagradável para o usuário.

A gravidade do problema de *aliasing* temporal é proporcional à intensidade do serrilhamento das sombras. Assim, se a reparametrização e o particionamento do mapa forem eficazes em eliminar o serrilhamento a menos de um pixel, o efeito torna-se imperceptível. Quando isso não é possível, como acontece na maioria dos casos, a aplicação do filtro PCF também contribui para dissimular as variações de qualidade quadro-a-quadro. Em nossa implementação, o *aliasing* temporal aparece de forma ligeiramente perturbadora apenas quando a luz e a câmera se aproximam da configuração paralela.

5.5. Resultado final

Após todo o estudo realizado, podemos recomendar um conjunto de técnicas para o aprimoramento do mapeamento de sombras na visualização de modelos CAD em tempo real. Recomendamos que o mapa de sombras seja gerado com *bias* constante, utilizando a reparametrização por n'_{LISPSM} generalizado, e realizando-se o particionamento adaptativo em profundidade com duas partições (ZP_2). Caso a capacidade de processamento gráfico do sistema seja suficiente ou taxas interativas sejam satisfatórias, pode-se substituir o *bias* constante pela

técnica de ponto médio *dual-depth*, bem como se utilizar três ou quatro partições em profundidade. Durante a renderização final, recomendamos a aplicação de um filtro PCF com núcleo de 3×3 a 5×5 baseado nos texels do mapa, amostragem adaptativa opcional e aproximação de amostras virtuais pelo plano tangente.

As figuras 30 a 33 ilustram progressivamente os resultados obtidos com a aplicação de cada técnica recomendada. As sombras ilustradas na Figura 33 foram obtidas com *bias* constante, particionamento ZP_2 e filtro PCF 5×5 sem amostragem adaptativa. Elas têm alta qualidade quando a luz e a câmera são perpendiculares, e qualidade ainda muito boa no caso em que são quase paralelas.

A Tabela 6 faz uma comparação de desempenho entre a técnica de mapeamento de sombras tradicional e a combinação das técnicas recomendadas (conforme descrita para a Figura 33). Pode-se perceber pela tabela que as técnicas de aprimoramento não causam necessariamente uma perda de eficiência, podendo inclusive produzir resultados mais rápidos que o algoritmo tradicional devido ao descarte em relação a volumes de visão ajustados a cada partição. A visualização se mantém em tempo real para uma fonte de luz e em taxas interativas para duas fontes.

Tabela 6 – Comparação de desempenho em uma GeForce 8800 GTS (em FPS), sempre com ponto-de-vista restrito

Luzes \ Técnica	Mapeamento Tradicional	Técnicas recomendadas
1 luz dinâmica	45	29 a 59
2 luzes dinâmicas	25	18 a 30

Tabela 7 – Comparação de desempenho em uma GeForce 6800 GT (em FPS) , sempre com ponto-de-vista restrito

Luzes \ Técnica	Mapeamento Tradicional	Técnicas recomendadas
1 luz dinâmica	33	9 a 12
2 luzes dinâmicas	18	5 a 7

Os resultados da Tabela 6 foram obtidos com a máquina mencionada no começo do capítulo, que possuía uma placa gráfica Nvidia GeForce 8800 GTS. Para completar nossos testes, realizamos medidas de eficiência também numa máquina com processador Intel Pentium 4 3GHz e placa gráfica Nvidia GeForce 6800 GT. Os resultados, apresentados na Tabela 7, exibem uma queda significativa de desempenho com a utilização das técnicas recomendadas.

Verificamos que tal queda se deve principalmente à utilização do filtro PCF, que produz um forte gargalo de rasterização nas passadas da renderização final. Esse fato, no entanto, não deve causar preocupação, já que a nova geração de placas gráficas exhibe uma tendência de grande aumento do poder de processamento de fragmentos.

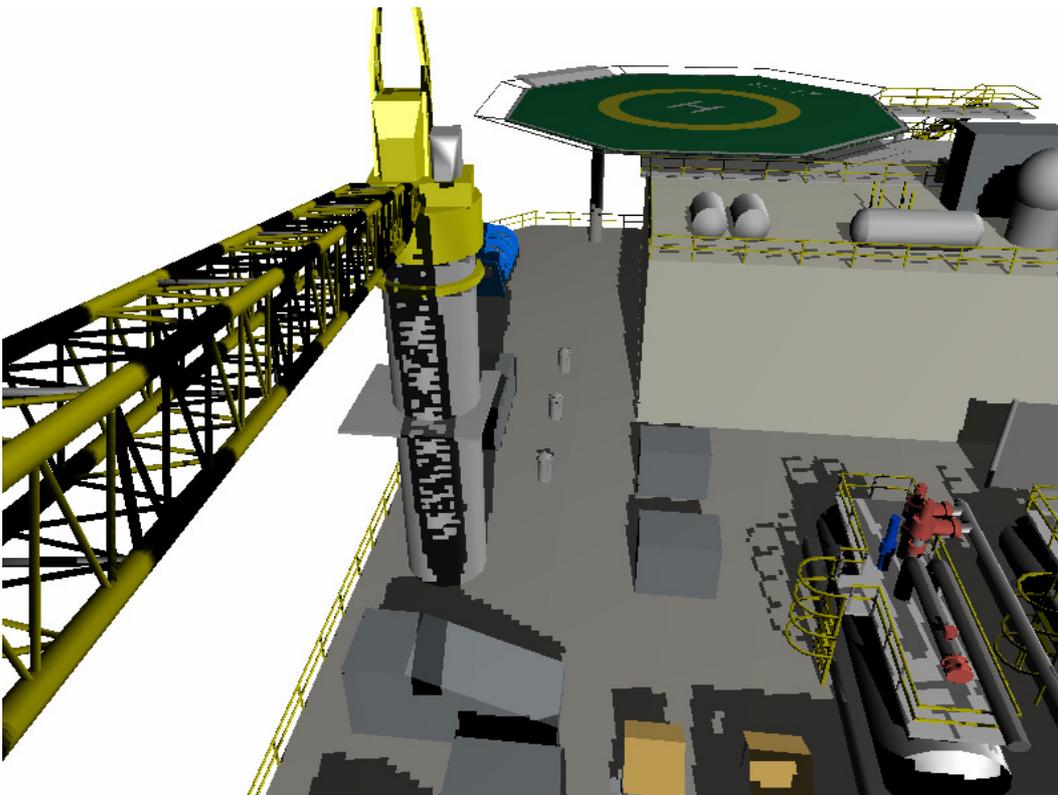


Figura 30 – Qualidades das sombras obtidas por mapeamento tradicional.
 Acima: situação próxima à ótima para a reparametrização; abaixo: situação próxima à de luz e câmera paralelas

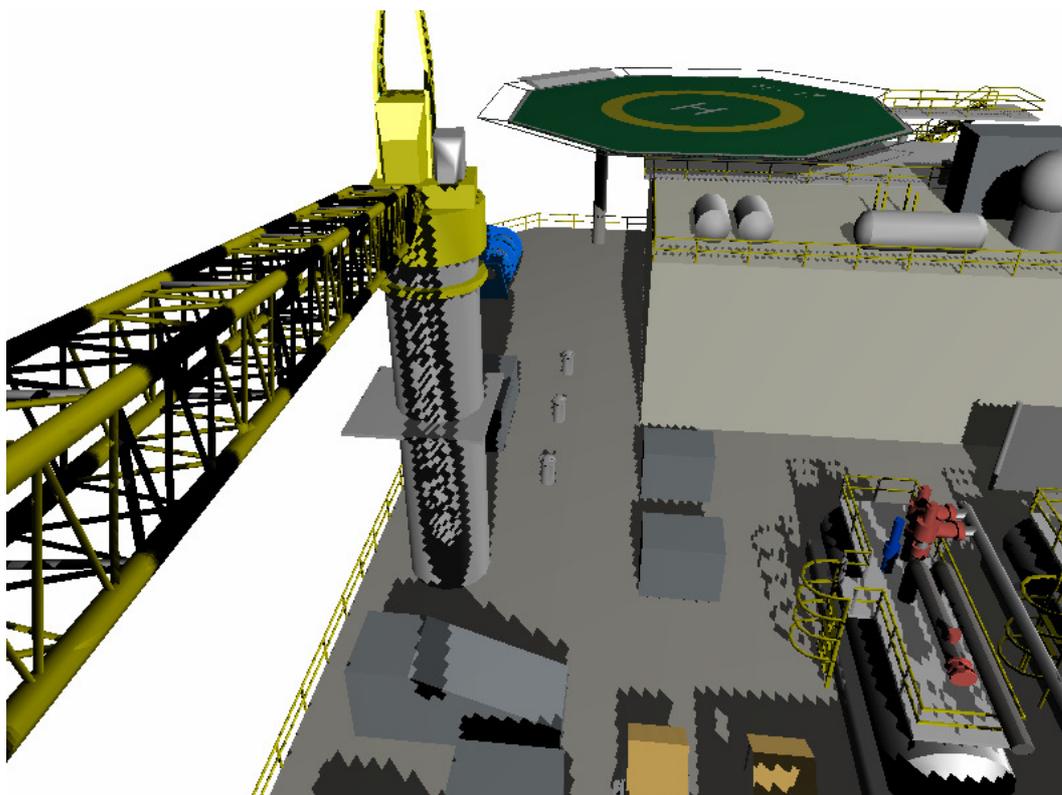
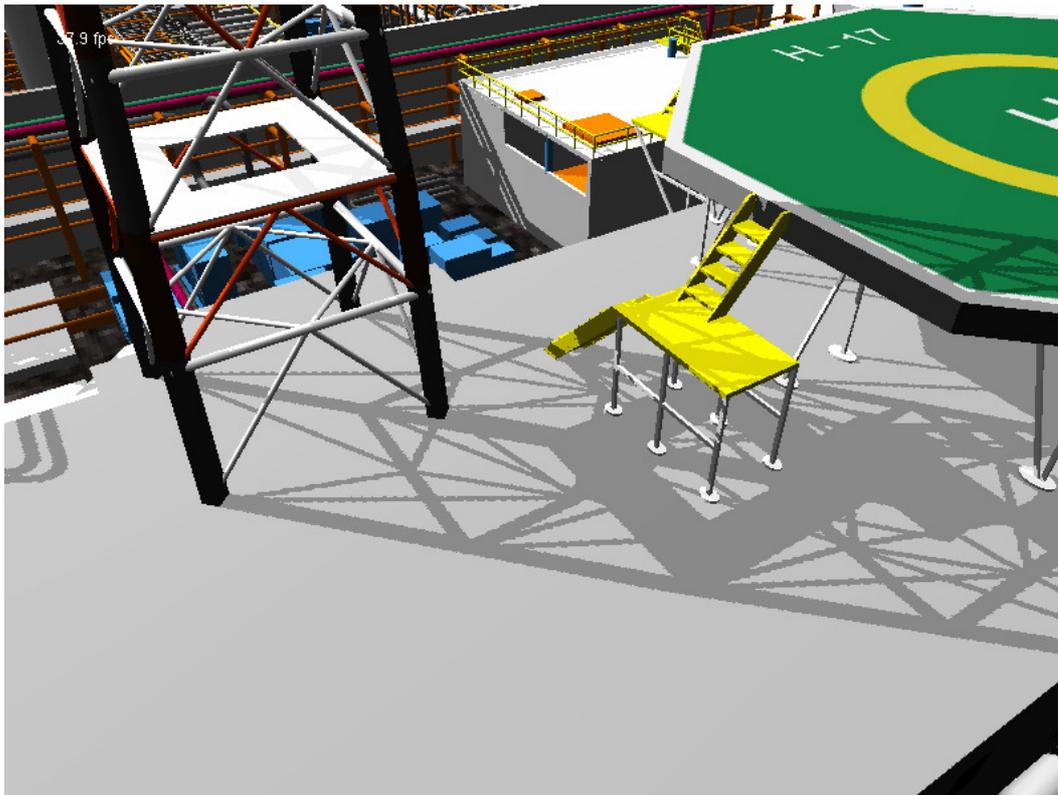


Figura 31 – Qualidades das sombras obtidas com reparametrização perspectiva.
 Acima: situação próxima à ótima para a reparametrização; abaixo: situação próxima à de luz e câmera paralelas

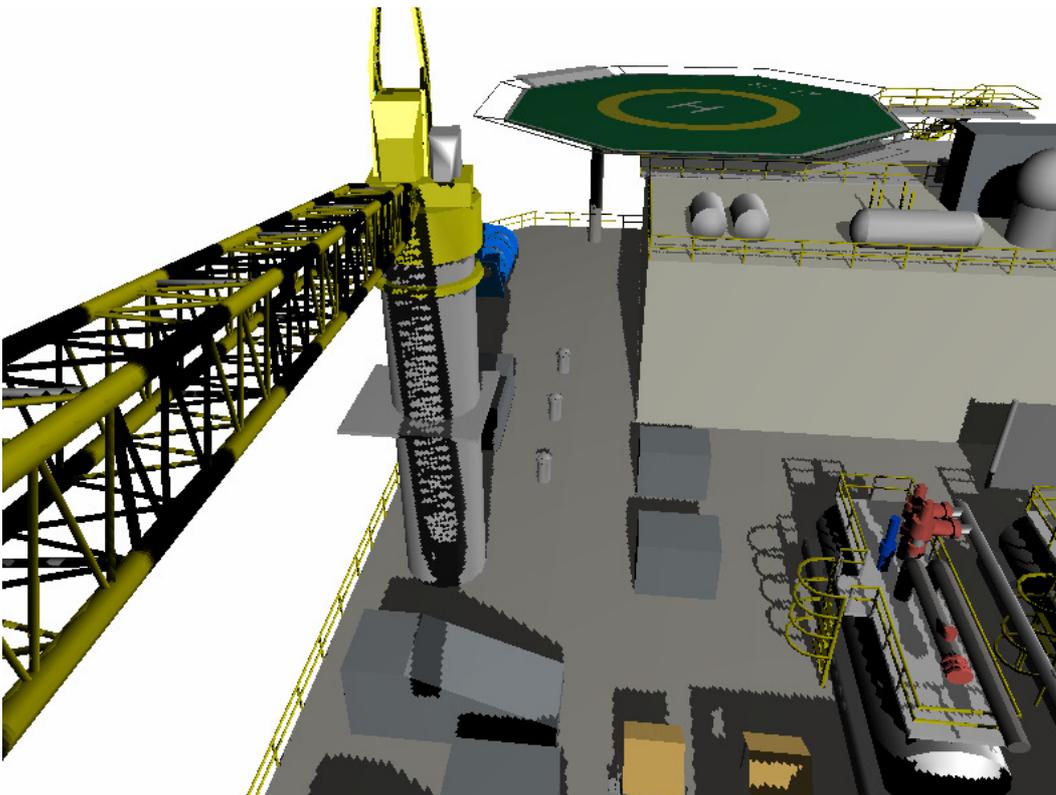
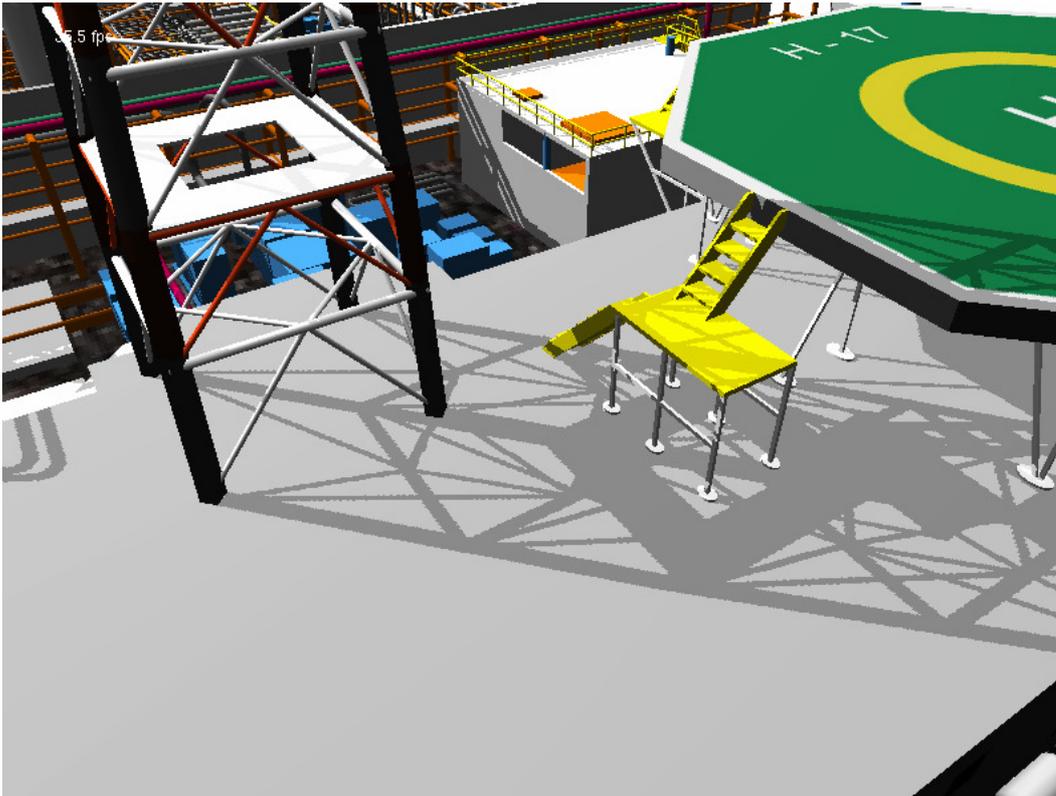


Figura 32 – Qualidades das sombras obtidas com reparametrização e particionamento em duas partições. Acima: situação próxima à ótima para a reparametrização; abaixo: situação próxima à de luz e câmera paralelas

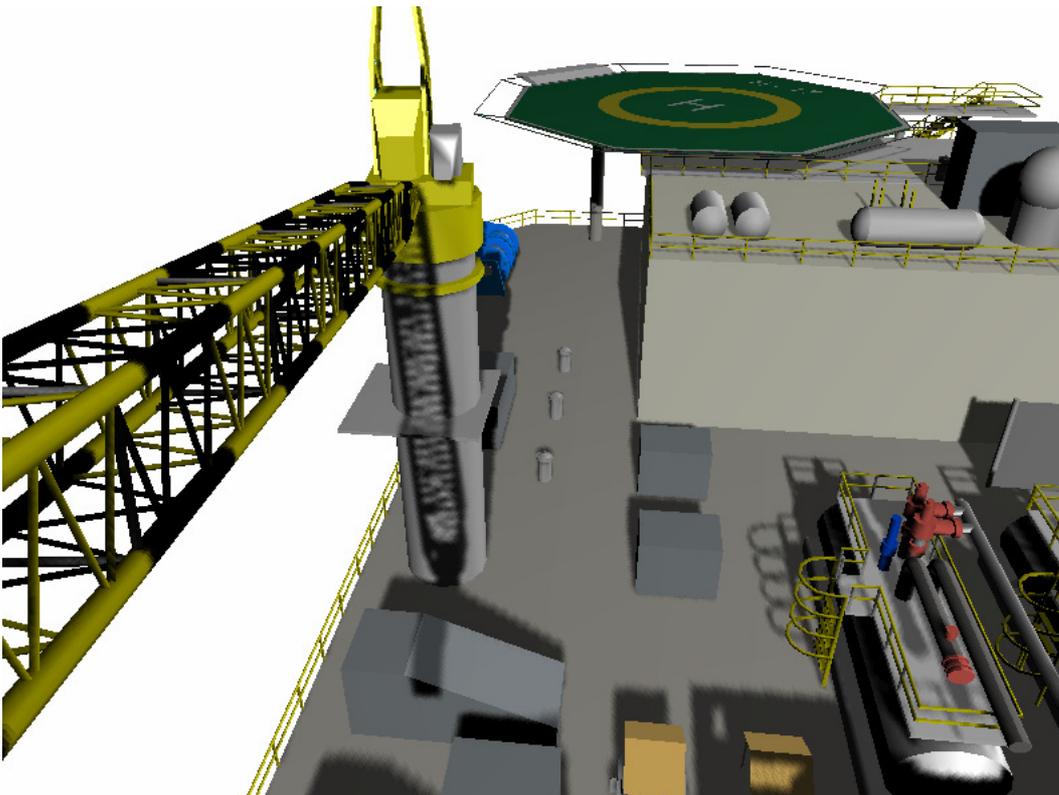
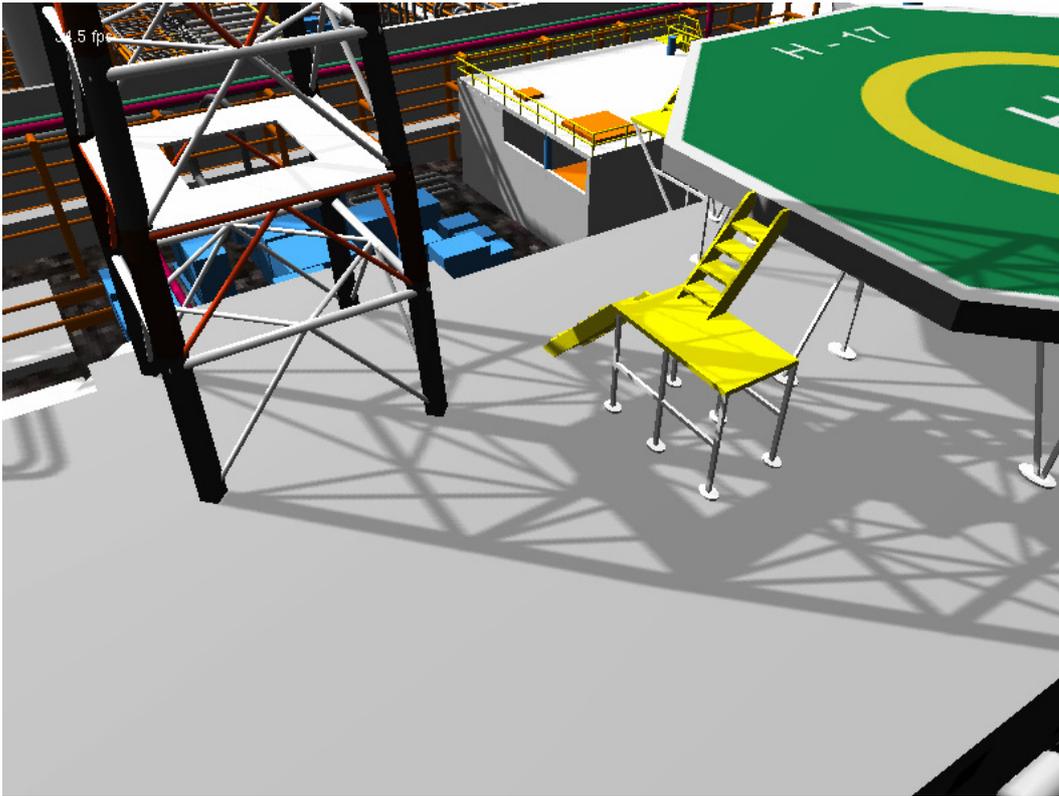


Figura 33 – Qualidades das sombras obtidas com a combinação de todas as técnicas recomendadas. Acima: situação próxima à ótima para a reparametrização; abaixo: situação próxima à de luz e câmera paralelas