

5

Qualidade dos Resultados

Neste capítulo apresentaremos alguns resultados do trabalho com a medida de qualidade da reconstrução por *TetraQuads* utilizando diferentes modelos e resoluções de malha de tetraedros. Os testes foram realizados em um iMac com processador Intel Core i5 de 2.9GHz, 8GB de RAM e placa de vídeo NVIDIA GeForce GTX 660M com memória de 512MB.

5.1

Construindo um Volume de Distâncias

Para avaliação da qualidade das superfícies reconstruídas com o *TetraQuad* nos inspiramos no método descrito por Cignoni et al (26) para comparação entre malha de triângulos em diferentes graus de refinamento. Como nossos pontos de entrada para a reconstrução são obtidos dos vértices de uma malha triangular, começamos por medir a proximidade entre o *TetraQuad* (S_1) e a malha (S_2) que é dada pela distância de Hausdorff:

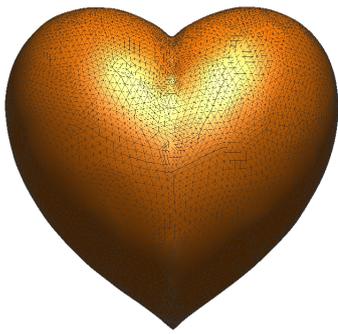
$$E(S_1, S_2) = \max_{p \in S_1} e(p, S_2), \quad (5-1)$$

onde $e(p, S)$ é a função que mede o mínimo da distância euclidiana (d) entre um ponto p e uma superfície S

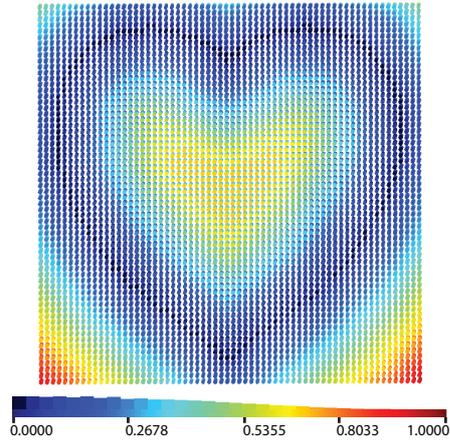
$$e(p, S) = \min_{p' \in S} d(p, p'). \quad (5-2)$$

Mas essa função de distância não é simétrica, $E(S_1, S_2)$ não é necessariamente igual à $E(S_2, S_1)$. Podemos então obter a distância entre as duas superfícies encontrando o máximo entre $E(S_1, S_2)$ e $E(S_2, S_1)$.

A nossa implementação tem a mesma intenção do método desenvolvido por Cignone apesar de não ter a mesma precisão e robustez. Construimos um volume de distâncias medidas entre a malha de triângulos de entrada e os *voxels* de um volume gerado amostrando regularmente o espaço do objeto. Para saber a distância de um ponto qualquer em relação a malha de entrada, basta consultar esse volume de distâncias. A figura 5.1(b) mostra um corte no volume de distâncias construído a partir da malha da figura 5.1(a).



5.1(a): Modelo *Cœur* representado com uma malha de triângulos.



5.1(b): Corte no volume de distâncias.

Figura 5.1: Construindo um volume de distâncias

Como a princípio não temos os pontos do *TetraQuad* para fazer a consulta, primeiramente é feita uma amostragem dos pontos da superfície reconstruída a partir do *buffer* de profundidade fornecido pela placa gráfica. Dividimos o volume de visão em fatias, cada uma delas é renderizada separadamente criando uma amostragem por pontos de toda a superfície a partir da leitura do *z-buffer* e suas coordenadas de tela (posição dos *pixels*). Por fim é realizada a transformação desses pontos para coordenadas de mundo e temos a nossa superfície amostrada.

O erro entre as duas superfícies é calculado através da média de todas as distâncias entre os pontos do *TetraQuad* e a malha de entrada:

$$Err = \frac{1}{n_p} \sum E(T, M),$$

onde T indica a malha de *TetraQuads*, M a malha de triângulos e n_p o número de pontos de entrada. Esse erro é mapeado em cores com as quais pintamos os pontos de entrada, a cor azul indica pontos com erros pequenos e a cor vermelha pontos resultantes de problemas na reconstrução. A figura 5.3 mostra as cores do histograma mapeadas sobre os pontos amostrados da superfície resultante da reconstrução (figura 5.2).

A função que construímos não é simétrica e definir uma função de distância a partir do *TetraQuad* é uma tarefa complexa. Desenvolvemos uma abordagem mais simples calculando as distâncias a partir do *buffer* de pontos que obtivemos da superfície. A distância $E(S_2, S_1)$, medida entre a malha e o

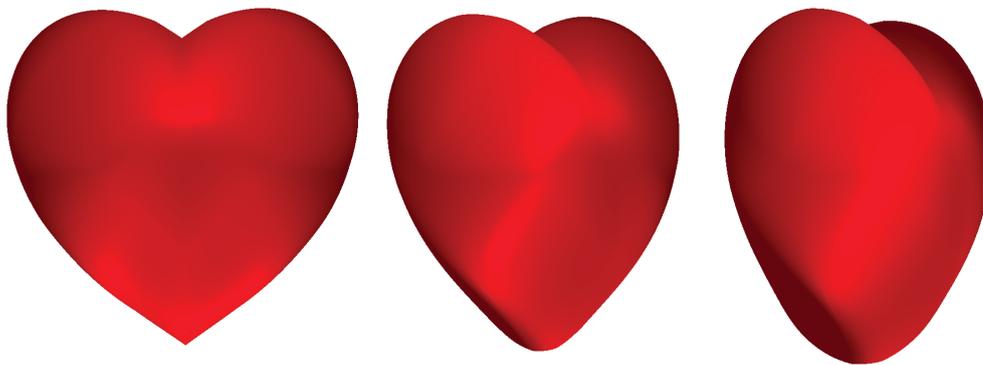


Figura 5.2: Resultado da reconstrução do modelo *Cœur* em uma grade de resolução $2 \times 2 \times 2$.

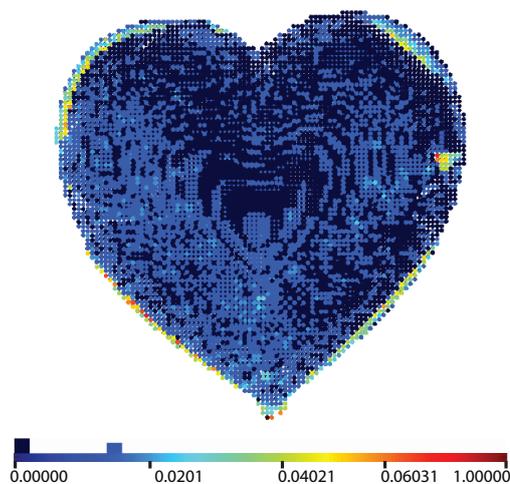


Figura 5.3: Visualização do histograma de pontos amostrados. A cor azul indica pontos com erros pequenos e a cor vermelha indica pontos resultantes de problemas na reconstrução.

TetraQuad, captura bem a qualidade e a completude da aproximação, enquanto a distância $E(S_1, S_2)$ permite detectar componentes espúrias. Como a presença dessas componentes depende muito do posicionamento da malha, e não tanto da qualidade do método de resolução de mínimos quadrados, usaremos a primeira distância no resto desse capítulo. A figura 5.4 mostra o histograma dos pontos de entrada construído consultando o volume de distâncias obtido a partir da nuvem de pontos do *TetraQuad*.

Iniciaremos a nossa análise pelos resultados obtidos com os diferentes métodos para resolução de mínimos quadrados apresentados no capítulo 4.

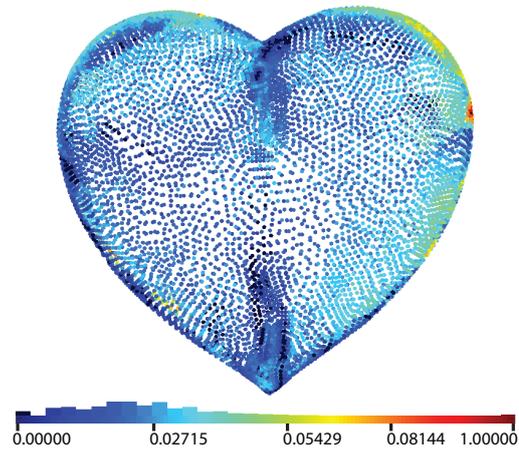


Figura 5.4: Histograma dos pontos de entrada.

5.2 Métodos de Resolução de Mínimos Quadrados

Na seção 4.9 já havíamos falado sobre a diferença entre as performances dos algoritmos de resolução de mínimos quadrados sem preocupação com a qualidade da reconstrução. A evolução do valor do erro de cada um dos métodos de acordo com o aumento da resolução da malha pode ser visto no gráfico da figura 5.5.

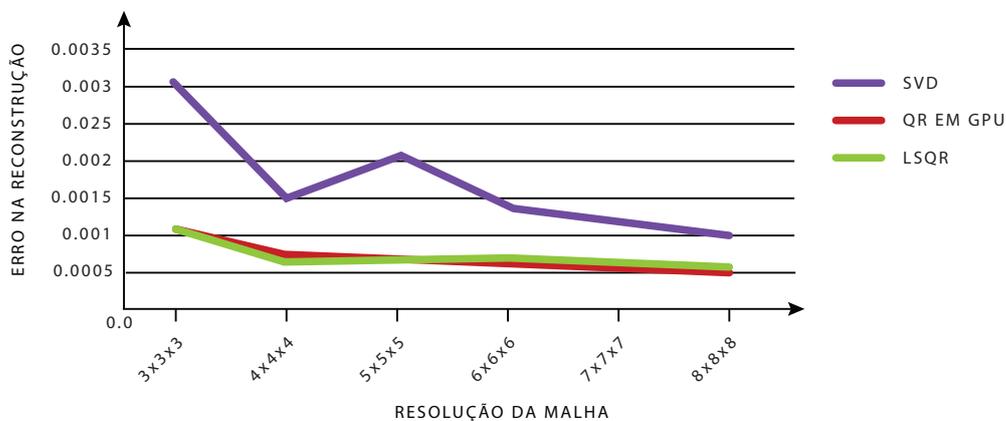


Figura 5.5: Gráfico de evolução do erro para cada um dos métodos.

O gráfico aponta uma proximidade grande entre os métodos QR e LSQR. Nas figuras 5.6, 5.7 e 5.8 podemos comprovar isso ao observar a diferença na qualidade da reconstrução entre os métodos para diferentes resoluções de malha.

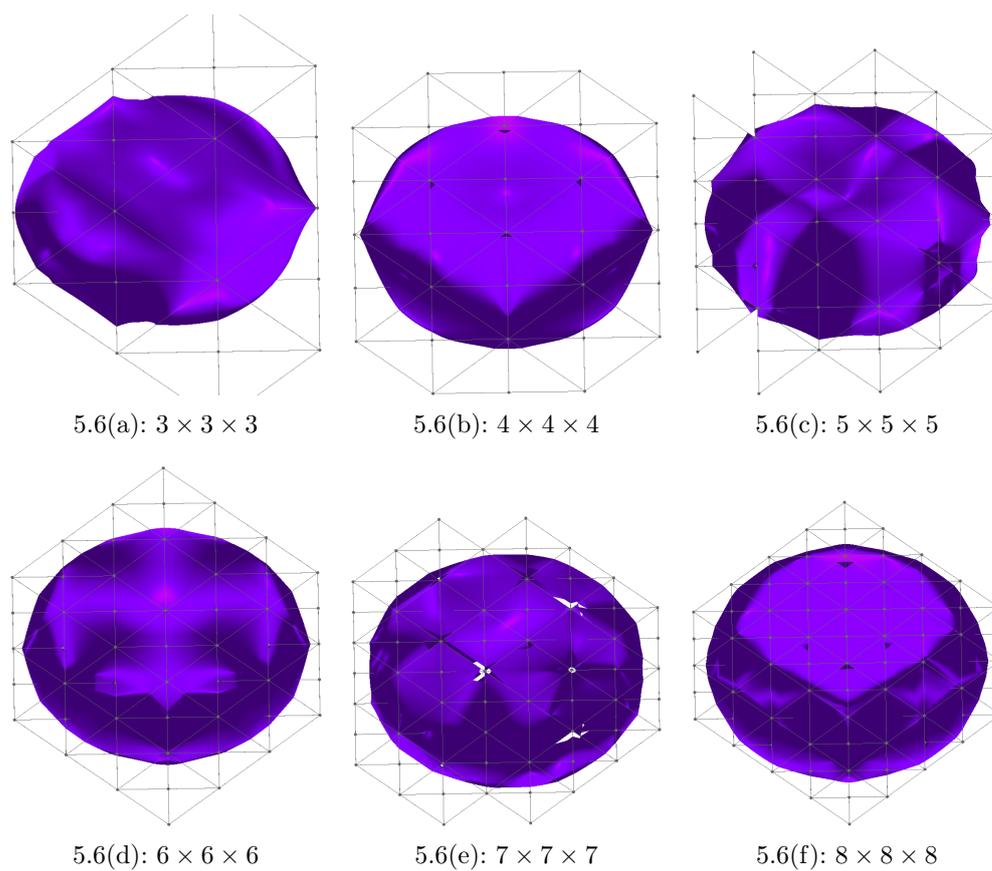


Figura 5.6: Resultados obtidos com SVD e diferentes resoluções de malha.

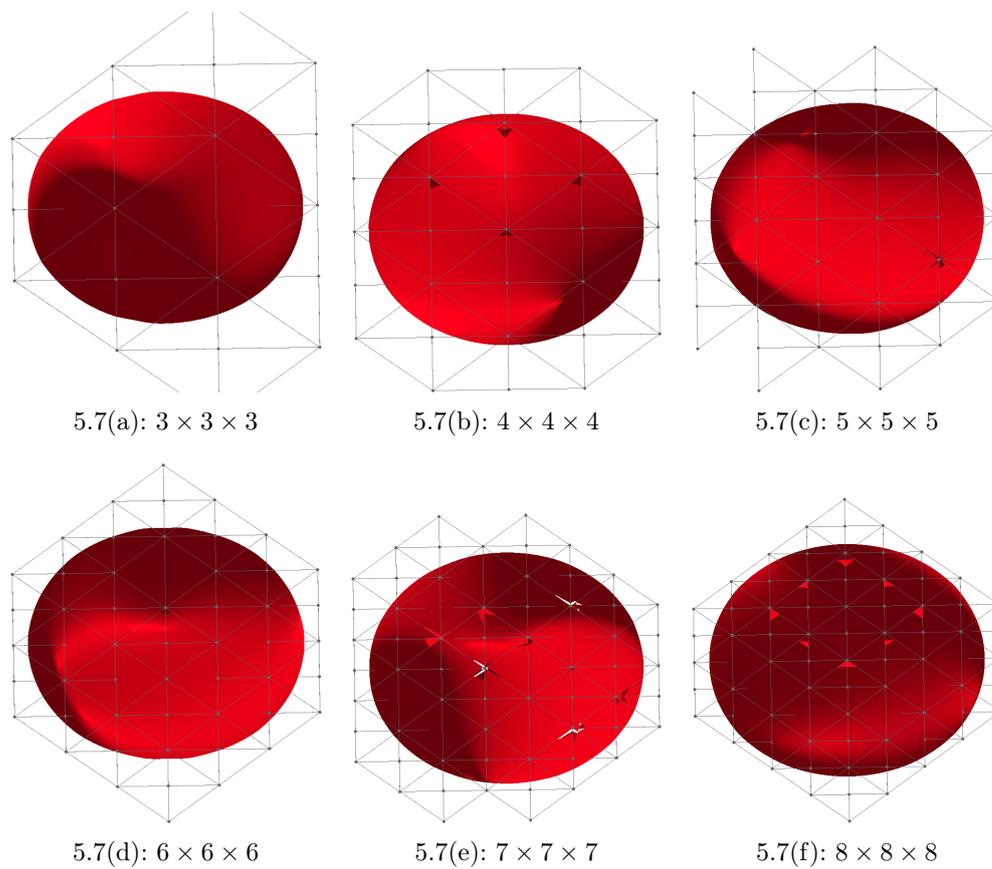


Figura 5.7: Resultados obtidos com o método QR em GPU e diferentes resoluções de malha.

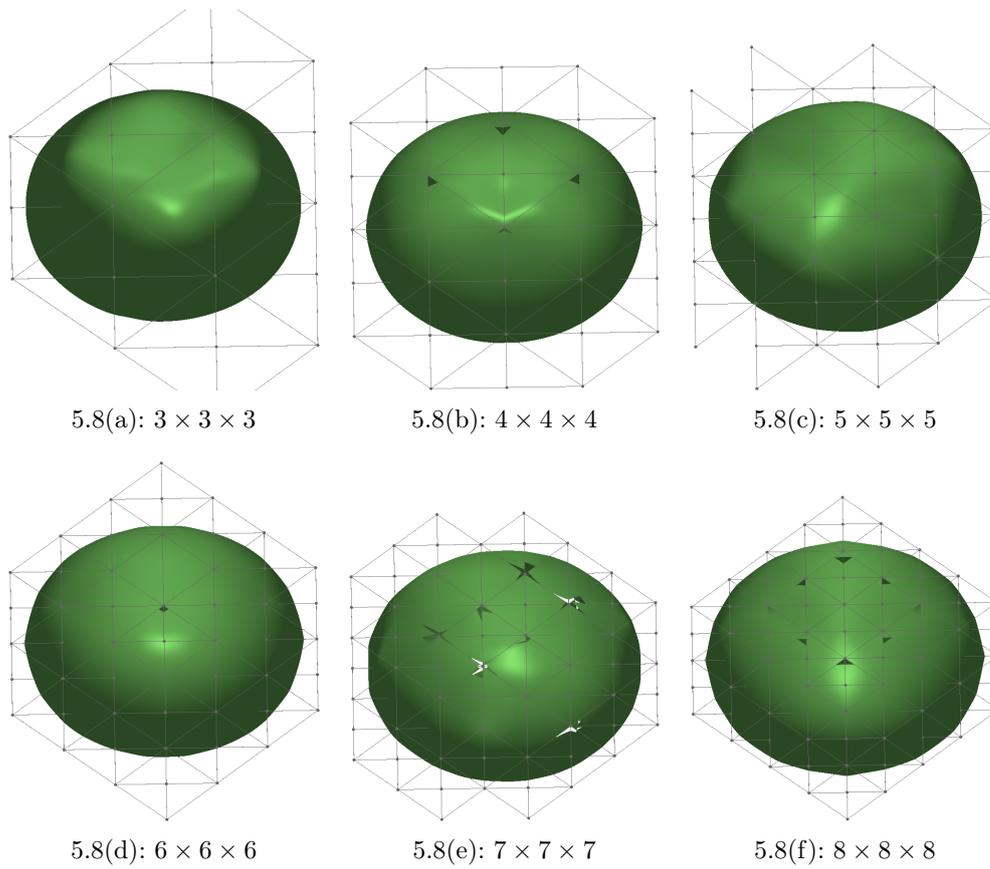


Figura 5.8: Resultados obtidos com o método LSQR e diferentes resoluções de malha.

5.3

Pesos das Equações

Como mencionamos no capítulo 4, o peso que é dado a cada uma das restrições na construção do sistema desempenha um papel importante no resultado final da reconstrução. Para comprovar esse fato, construímos três gráficos relacionando os pesos ao erro médio na reconstrução que definimos no início do capítulo. Em todas as situações, utilizamos uma grade de resolução $4 \times 4 \times 4$ e como malha de entrada o modelo *Bunny* com 8971 vértices. Definimos um valor inicial para o peso dado a cada uma das restrições e fomos variando o valor de cada um deles separadamente. O sistema foi construído utilizando ao mesmo tempo as equações das normais, a redução dos coeficientes de grau 3, diferentes superfícies de nível, o coeficiente J livre e o ajuste na posição dos vértices. Nosso objetivo é verificar qual o melhor cenário para a reconstrução do modelo com a finalidade de minimizar imperfeições.

A primeira variação de valores foi testada com o peso dado as equações das normais e pode ser observado no gráfico da figura 5.9, junto com uma imagem do melhor resultado. O menor erro foi alcançado para o peso 0.044 e o maior para o valor 0.1. Os resultados para as outras restrições, redução dos altos coeficientes e diferentes superfícies de nível, podem ser vistos nas figuras 5.10 e 5.11. Vale notar que o melhor resultado foi obtido na resolução intermediária.

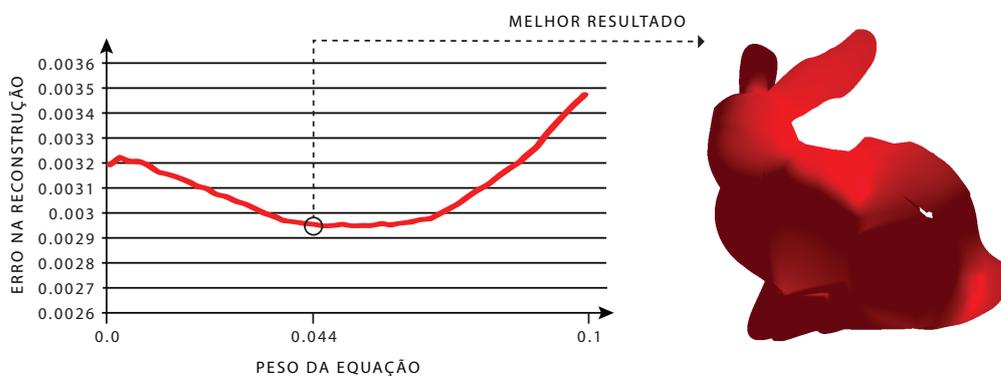


Figura 5.9: Comportamento do erro ao variarmos o peso das equações das normais.

Agora que conhecemos bons valores de pesos para as equações, utilizamos essas informações para reconstruir o modelo *Bunny* com 27734 vértices de entrada em diferentes resoluções de grade. Nas figuras 5.12, 5.13 e 5.14

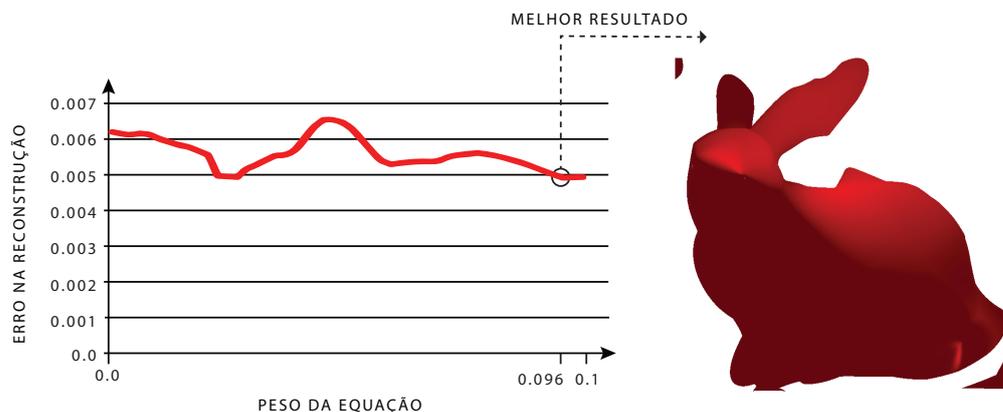


Figura 5.10: Variando o peso das equações de redução da influência dos coeficientes de grau 3.

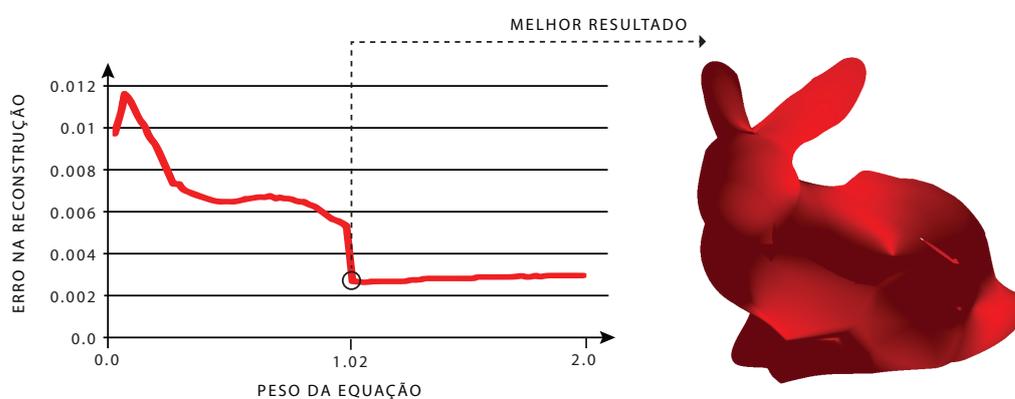


Figura 5.11: Variando o peso das equações das superfícies de nível.

utilizamos respectivamente grades de resolução $5 \times 5 \times 5$, $8 \times 8 \times 8$ e $10 \times 10 \times 10$. Os valores dos erros medidos para cada um dos casos aparecem nas imagens.

O mesmo processo foi utilizado para reproduzir os resultados das figuras 5.15 e 5.16 tendo como entrada o modelo *Cassini* com 32167 vértices em malhas de resolução $4 \times 4 \times 4$ e $6 \times 6 \times 6$. O melhor resultado ocorreu para o último caso, onde podemos observar também uma melhoria na suavidade do campo de normais.

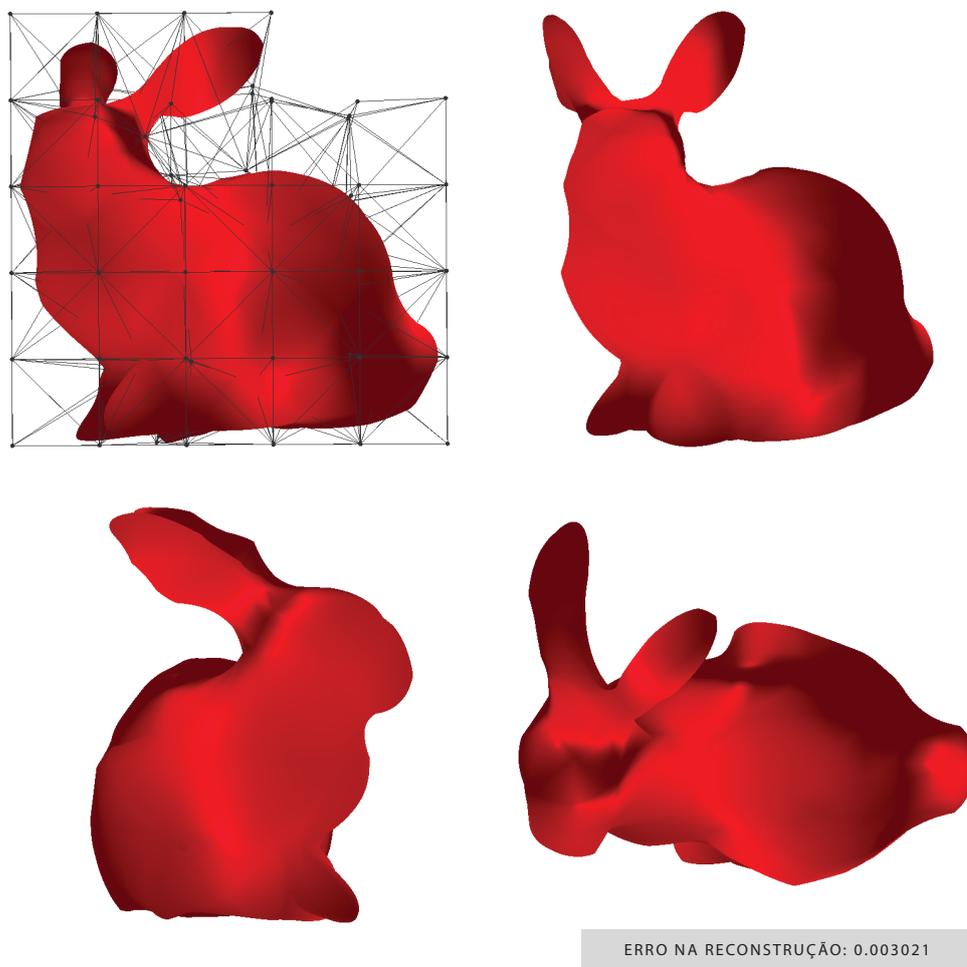


Figura 5.12: Bunny em uma malha de resolução $5 \times 5 \times 5$.

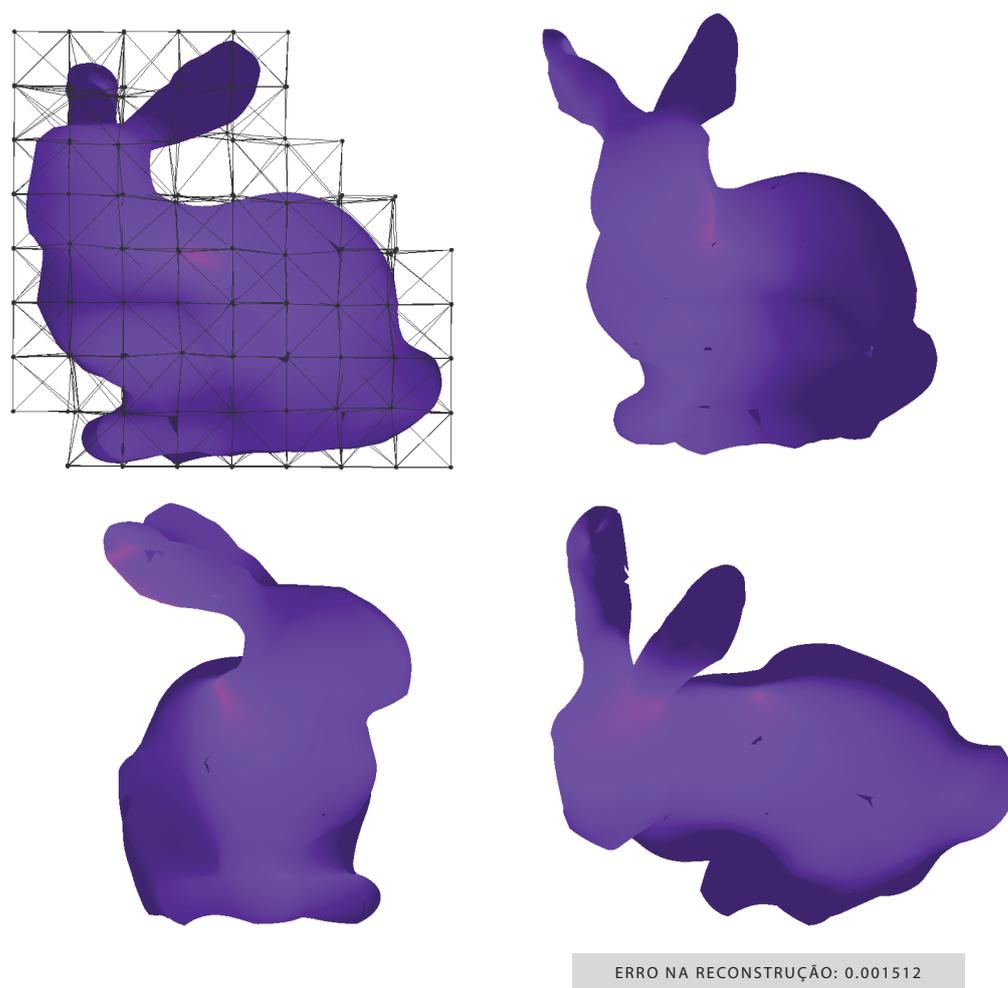


Figura 5.13: Bunny em uma malha de resolução $8 \times 8 \times 8$. Esse é o melhor resultado obtido com o modelo.

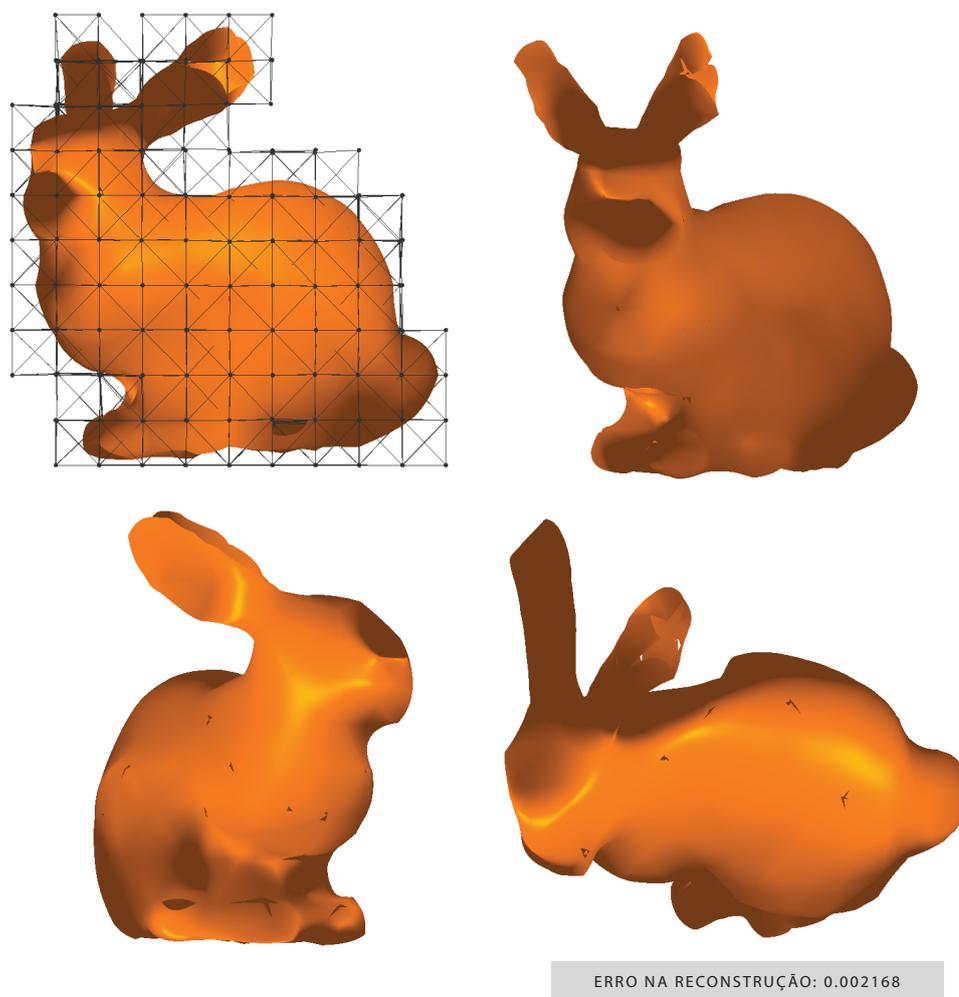


Figura 5.14: Bunny em uma malha de resolução $10 \times 10 \times 10$.

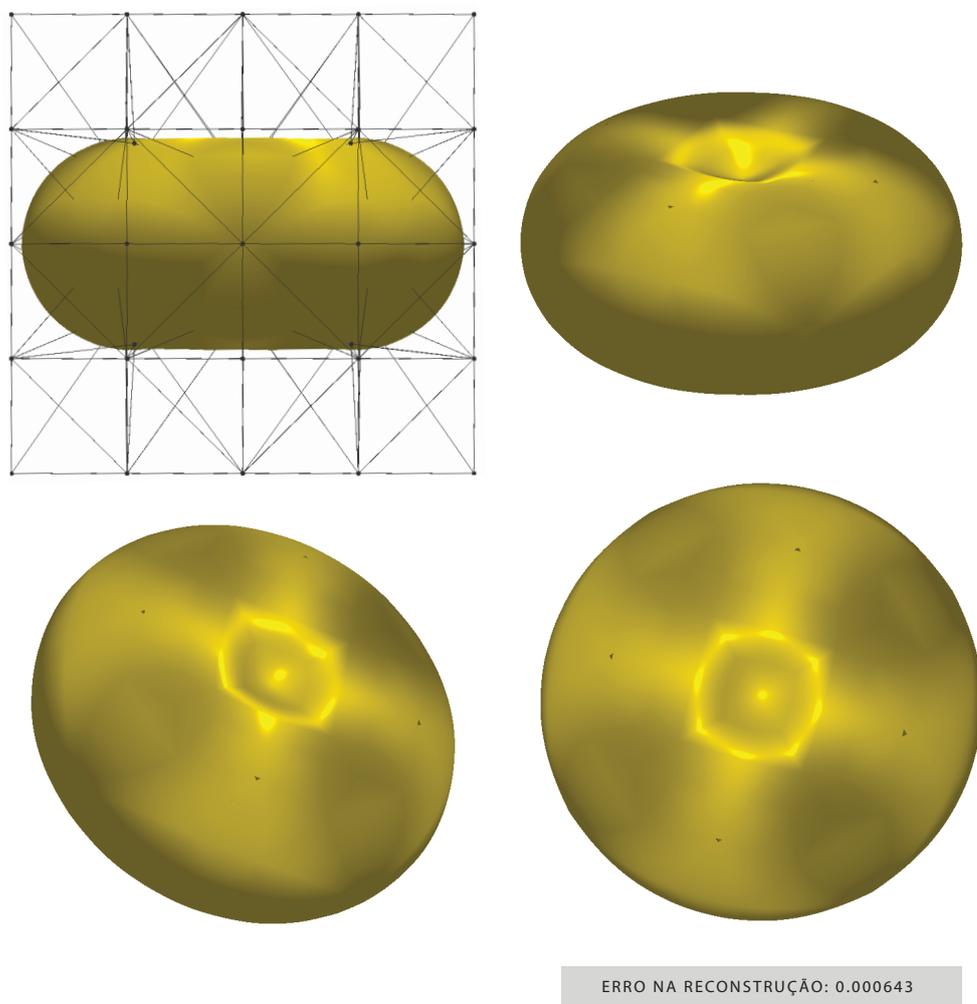
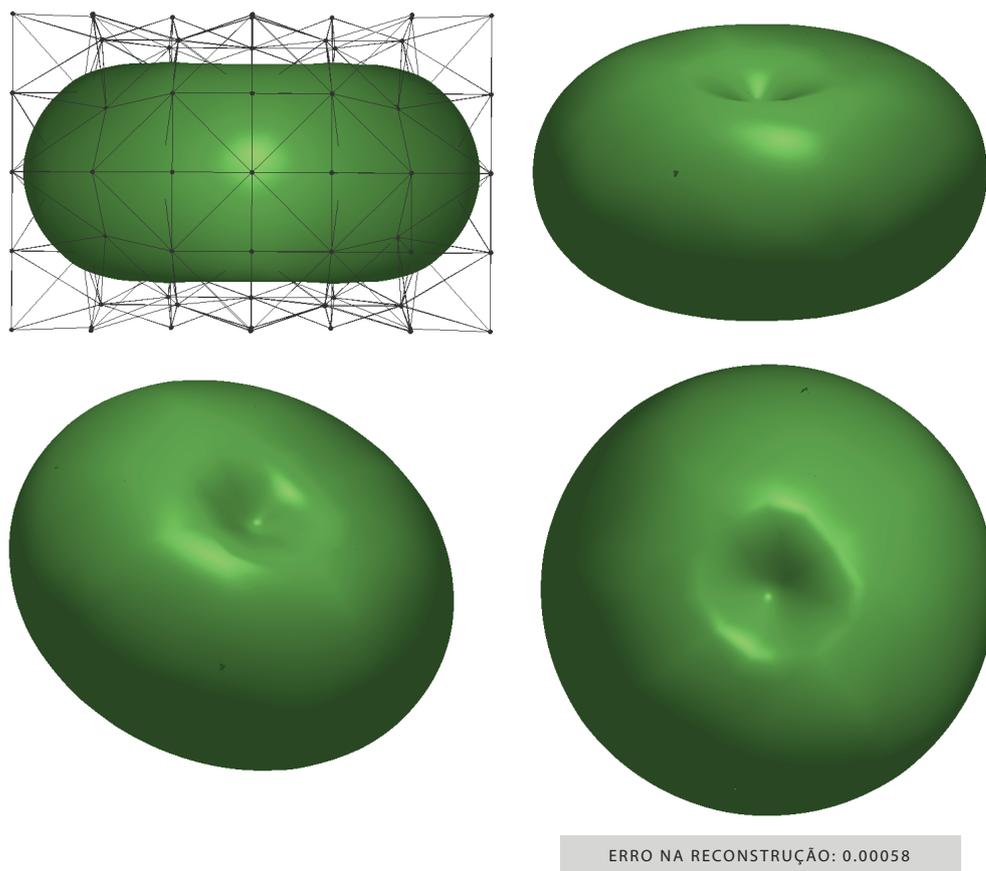


Figura 5.15: Cassini em uma malha de resolução $4 \times 4 \times 4$. Nesse resultado podem ser observadas irregularidades no campo de normais.



PUC-Rio - Certificação Digital Nº 1012858/CA

Figura 5.16: Cassini em uma malha de resolução $6 \times 6 \times 6$. Esse é o melhor resultado obtido com o modelo.