

Projeto 2: Maximum Intensity Projection

Marcos Teixeira

23 de julho de 2018

1 Descrição

Maximum Intensity Projection (MIP) é um método de visualização de volume para dados 3D que projeta no plano de visualização os voxels com intensidade máxima que se enquadram no caminho de raios paralelos traçados desde o ponto de vista até o plano de projeção. Observe que isso implica que duas renderizações MIP de pontos de vista opostos são imagens simétricas. Essa técnica é computacionalmente rápida (se implementada de maneira rigorosa), mas os resultados 2D não fornecem um bom senso de profundidade dos dados originais. Para melhorar o sentido de 3D, geralmente são renderizadas animações de vários quadros MIP em que o ponto de vista é ligeiramente alterado de um quadro para outro, criando assim a ilusão de rotação. Isso ajuda a percepção do usuário a encontrar as posições 3D relativas dos componentes do objeto.

Para este projeto a tarefa é implementar o MIP de maneira eficiente. Nas seções posteriores apresentamos os detalhes deste método, bem como os resultados em algumas imagens 2D.

2 Maximum Intensity Projection

Maximum Intensity Projection (MIP) [1] consiste em projetar o voxel com o maior valor de brilho em todas as vistas do volume em uma imagem 2D. Utilizamos esse tipo de projeção para destacarmos as partes importantes da imagem. Uma aplicação recorrente desta técnica é na identificação de nódulos pulmonares em câncer de pulmão nos programas de tomografia computadorizada. Para aumentar o senso de tridimensionalidade, animações são, muitas vezes, renderizadas com vários frames de MIP em que os pontos de vista são ligeiramente diferentes, criando uma ilusão de rotação [2].

O pipeline de execução do MIP pode ser expresso da seguinte maneira:

- i Criar a imagem de saída $R(D, D)$
- ii Criar a matriz de transformação M_{-1}
- iii Definição das faces da cena

iv Aplicar M_{-1} na matriz normal

v Para cada ponto do plano de visualização, faça

(a) Aplicar M_{-1}

(b) Se houver intersecção

i. Calcular os λ válidos que dam origem a $p1$ e pn

ii. Aplicar o DDA para percorrer os pontos entre $p1$ e pn em direção a cena

A. Interpolar linearmente os pontos válidos e obter o valor de máxima intensidade

(c) Caso contrário, $R = 0$

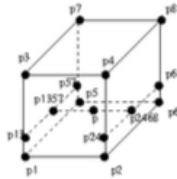
vi Retornar R

, onde a matriz $M_{-1} = T(N_x/2, N_y/2, N_z/2) * R_x(theta_x) * R_y(theta_y) * T(-D/2, -D/2, -D/2)$ com $D = \sqrt{nx^2 + ny^2 + nz^2}$.

Para cada uma das 6 faces, realizamos os cálculos apresentados na equação abaixo. Para cada um dos lambdas obtidos, nós pegamos o menor e o maior válidos para conseguir obter $p1$ e pn . O valor de menor lambda será atribuído a equação da reta para encontrar $p1$ e o maior lambda para encontrar pn . Note que é necessário garantir que $(\vec{n} \cdot \vec{n}_j)$ não sejam 0 para que os testes das equações acima funcionem.

$$\lambda_j = ((c_j - p_0) \cdot \vec{n}_j) / (\vec{n} \cdot \vec{n}_j),$$

$$p' = p_0 + \lambda \vec{n}$$



$$\begin{aligned} I_p &= (x_p - x_{p1357})I_{p2468} + (x_{p2468} - x_p)I_{p1357} \\ I_{p2468} &= (z_p - z_{p24})I_{p68} + (z_{p68} - z_p)I_{p24} \\ I_{p1357} &= (z_p - z_{p13})I_{p57} + (z_{p57} - z_p)I_{p13} \end{aligned}$$

Posteriormente, o algoritmo DDA será utilizado para encontrar todos os pontos da linha entre $p1$ e pn . Além de encontrar esses pontos, o DDA acumula os brilhos dos respectivos pontos que for passando e, no caso do MIP, guarda para retornar ao fim o valor do máximo brilho. Como as coordenadas (x, y, z) são reais, duas estratégias foram adotadas para a resposta do DDA: aplicar um *ROUND* das coordenadas e aplicar interpolação linear.

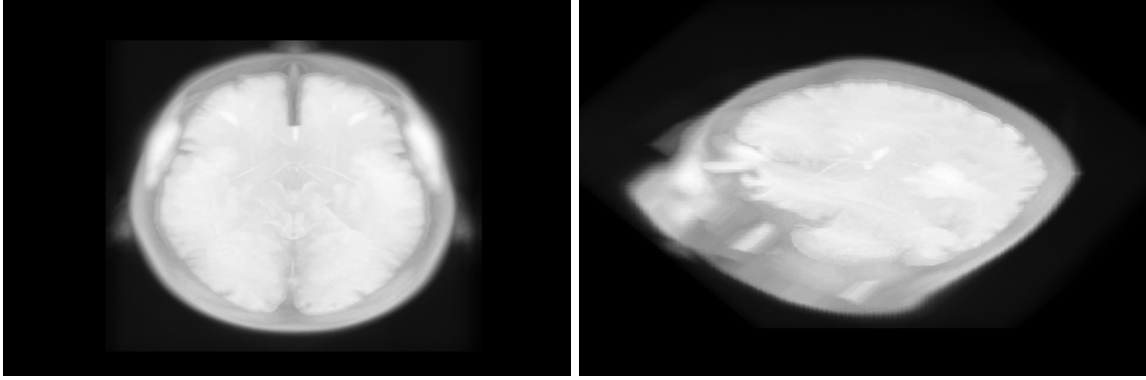


Figura 1: Resultados para imagem brain.csn. À esquerda, a imagem em (0,0) e à direita rotacionada em (45,45)

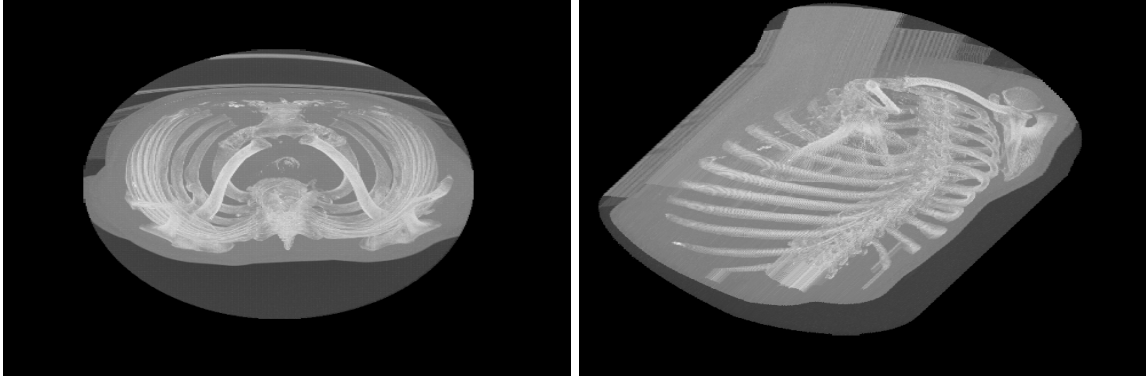


Figura 2: Resultados para imagem lungs.csn. À esquerda, a imagem em (0,0) e à direita rotacionada em (45,45)

Esperamos que o resultado utilizando interpolação linear possua menos buracos que na abordagem que só realiza o *ROUND*, dado que de um lado estaremos fazendo uma estimativa linear do brilho do pixel e do outro lado estaremos apenas truncando o valor das coordenadas. O método de interpolação adotado está descrito na Figura 2 e está implementado na função *iftImageValueAtPoint()* da biblioteca *ift*.

3 Resultados

Nesta seção apresentamos os resultados obtidos em algumas imagens de teste. As imagens de saída são da forma (D, D) , onde D é igual a $D = \sqrt{nx^2 + ny^2 + nz^2}$, representando a diagonal da imagem. Para padronizar as figuras nesse relatório, dado que esse D depende de cada imagem, nós limitamos a altura para 5cm e a largura para 7.5cm. Uma normalização de (0,255) é feita após a finalização do algoritmo MIP, para melhor visualização. Inicialmente, o plano de visualização é reposicionado em $z = D/2$, para afastar um pouco da visão do

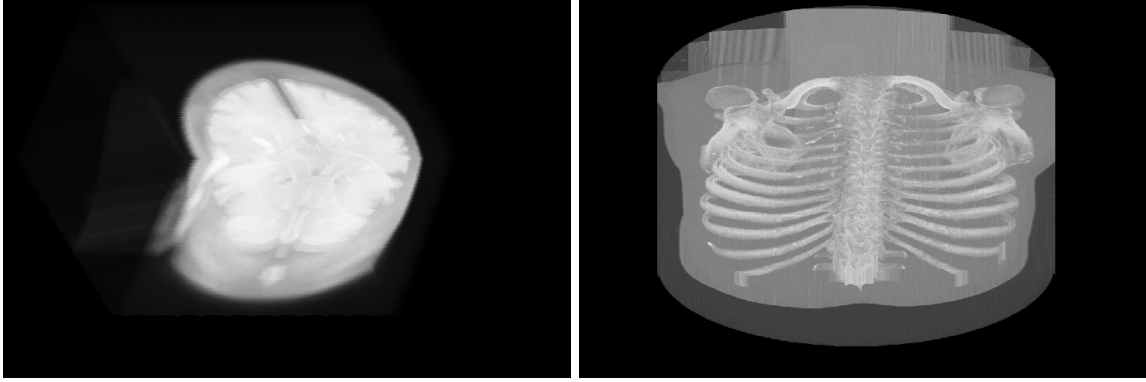


Figura 3: À esquerda, a imagem brain.csn em (50,20) e a imagem lungs.csn à direita rotacionada em (50,0)

observador.

Na Figura 2 mostramos os resultados do MIP em duas imagens de test para alguns valores de rotação em X e Y. Este resultado já está com interpolação linear. Os experimentos sem interpolação ficam bem parecidos com os usando interpolação, portanto nós optamos por suprimí-los deste relatório. A única diferença significativa foi em relação ao tempo de execução, dado que a abordagem com interpolação linear é mais custosa e, consequentemente, toma mais tempo de processamento. Mais resultados estão apresentados na Figura 3.

4 Conclusão

Neste trabalho foi apresentado uma implementação para o algoritmo de volume rendering chamado Maximum Intensity Projection. Através da teoria vista em sala de aula, fizemos uma modelagem que levasse em conta todos os passos do algoritmo de forma simples e objetiva. Para o DDA, inicialmente adotamos a interpolação e o *ROUND* como forma de obtenção dos brilhos intermediários na reta entra p_1 e p_n . Os resultados utilizando interpolação ficaram parecidos com os utilizando apenas *ROUND*, tirando o tempo de processamento maior devido as várias comparações necessárias. Foram feitos testes com diversas combinações de θ_x e θ_y e os resultados foram bons para todos.

Apesar das dificuldades iniciais com a implementação, o trabalho fluiu de forma satisfatória. Do ponto de vista conceitual, esse trabalho utiliza muito do que foi usado para o Projeto 1 (Transformada Rápida de Radon), o que facilitou um pouco o trabalho. De qualquer forma, foi um aprendizado relevante para concretizar a teoria vista em sala.

Algumas observações técnicas para a implementação. Foi necessário utilizar a biblioteca compilada para Mac OS disponibilizada pelo monitor para o código. Além disso, mudamos também a versão do gcc no Makefile para o gcc-7, para poder funcionar.

Referências

- [1] S. Napel, M. Marks, G. Rubin, M. Dake, C. McDonnell, S. Song, D. Enzmann, and R. Jeffrey Jr, “Ct angiography with spiral ct and maximum intensity projection.,” *Radiology*, vol. 185, no. 2, pp. 607–610, 1992.
- [2] H. H. Kondo, “Miprenderer uma ferramenta para realizar projeções de máxima intensidade de angiogramas utilizando vtk e qt.,” 2010.