

IMPLEMENTAÇÃO DE UM MOTOR GRÁFICO 3D COM AUXÍLIO VISUAL À DALTONISMO

João Vitor Miguel¹
Erick Stacy Gagliardi²
Fabio Pignelli²

RESUMO

Este trabalho apresenta o desenvolvimento de um motor gráfico 3D acessível para usuários com daltonismo, implementado sem o uso de bibliotecas gráficas externas. A proposta combina conceitos de computação gráfica, percepção visual e acessibilidade, com foco em filtros de simulação para protanopia, deuteranopia e tritanopia, aplicados diretamente no pipeline de renderização. Utilizando ray tracing e iluminação baseada no modelo de Phong, o sistema foi projetado para operar inteiramente em *Central Processing Unit* (CPU), garantindo controle total sobre o processo de renderização. Os testes revelaram uma sobrecarga computacional média inferior a 10%, mantendo a viabilidade mesmo em hardware convencional. A eficácia dos filtros foi avaliada comparativamente a soluções presentes em motores comerciais como Unity e Unreal Engine 5, demonstrando resultados qualitativos satisfatórios. A pesquisa contribui com um framework replicável para inclusão de acessibilidade visual em motores gráficos educacionais e profissionais, abrindo caminhos para futuras otimizações com suporte a GPU e pós-processamento avançado.

Palavras-chave: Motor-Gráfico. Raytracing. Daltonismo. Acessibilidade. Computação Gráfica.

ABSTRACT

This work presents the development of a 3D graphics engine designed to enhance accessibility for users with color vision deficiencies, implemented without relying on external graphics APIs. The approach integrates concepts from computer graphics, visual perception, and accessibility, focusing on color transformation filters for protanopia, deuteranopia, and tritanopia applied directly within the rendering pipeline. Using ray tracing and Phong lighting, the system operates entirely on the Central Processing Unit (CPU), providing full control over the rendering process. Performance tests showed an average computational overhead of less than 10%, confirming feasibility even on consumer-grade hardware. The effectiveness of the implemented filters was evaluated through comparisons with existing accessibility features in commercial engines such as Unity and Unreal Engine 5, demonstrating qualitatively satisfactory results. This research contributes a replicable framework for embedding visual accessibility into educational and professional rendering engines, laying the groundwork for future enhancements using GPU acceleration and advanced post-processing techniques.

Keywords: Graphics Engine. Ray Tracing. Color Blindness. Accessibility. Computer Graphics.

1. INTRODUÇÃO

O desenvolvimento de motores gráficos 3D tem se mostrado uma prática fundamental para diversas aplicações, como jogos, simulações e modelagem tridimensional (SHIRLEY; MARSCHNER, 2020). Essas ferramentas permitem a criação de ambientes virtuais complexos, oferecendo interatividade e realismo visual. No entanto, uma questão promissora nesse contexto é a acessibilidade, com diversos trabalhos recentes explorando seu

¹ Estudante de Engenharia de Software no Centro Universitário Cidade Verde - UniCV.

² Docente no Centro Universitário Cidade Verde - UniCV.

potencial, especialmente no que diz respeito a usuários com daltonismo (NAPOLI; CHIASSON, 2018).

A implementação de motores gráficos com recursos de acessibilidade para portadores de daltonismo baseia-se em conceitos fundamentais da computação gráfica, percepção visual e processamento digital de imagens. Os princípios matemáticos do ray tracing são recomendados na literatura como uma técnica fundamental para simulação de iluminação e sombreado (SHIRLEY; MARSCHNER, 2020). Esses princípios incluem o traçado de raios primários a partir da câmera virtual e cálculos precisos de interseção com objetos na cena.

No campo da oftalmologia, os distúrbios de visão cromática afetam significativamente a população, com prevalência de aproximadamente 8% em homens e 0,5% em mulheres (ZHU e MAO, 2021). Essa pesquisa adota especificamente as matrizes de transformação, que permitem simular digitalmente as limitações de percepção de cores características da protanopia, deuteranopia e tritanopia (BRETTEL; VIÉNOT; MOLLON, 1997). Essa técnica representa atualmente o padrão ouro para simulações de daltonismo em aplicações computacionais.

Diante disso, este projeto teve como objetivo desenvolver um motor gráfico 3D sem a utilização de bibliotecas API gráficas, incorporando recursos visuais que ampliem a acessibilidade para usuários com diferentes tipos de limitações na percepção de cores, a fim de melhorar sua experiência em aplicações gráficas.

A motivação se justifica na apresentação de uma solução acessível em motores gráficos, impactando a experiência de uma parcela significativa de usuários. A implementação de filtros adaptativos contribui para a inclusão digital, além de servir como base para futuras pesquisas em computação gráfica acessível (NAPOLI; CHIASSON, 2018).

Assim, este trabalho apresenta uma solução prática para um problema relevante, combinando conhecimentos em programação gráfica e design inclusivo. Os resultados obtidos serão, potencialmente, aplicados em jogos, softwares de modelagem e outras ferramentas que dependam de representações visuais precisas, beneficiando tanto usuários com daltonismo quanto desenvolvedores interessados em acessibilidade.

2. MATERIAIS E MÉTODO

O desenvolvimento do motor gráfico seguiu uma abordagem incremental, iniciando pela implementação básica dos algoritmos de ray tracing e progressivamente incorporando

funcionalidades avançadas. O sistema foi projetado para ser executado inteiramente em ambiente de *Central Processing Unit* (CPU), ou seja, todas as operações foram realizadas por processamento geral, sem recorrer a placas gráficas ou aceleradores dedicados, utilizando de linguagem de programação orientada a objetos C++. Os arquivos foram distribuídos entre classes de materiais e objetos, tais como metal, vidro e esfera, e classes para controle de criação de raios de colisão permitindo controle total sobre a renderização sem dependência de bibliotecas gráficas externas (DIRECT3D, 2025). O gerenciamento de cena incluiu a modelagem matemática de objetos primitivos como esferas e planos, organizados em um sistema de coordenadas mundial com suporte a transformações geométricas complexas.

Utilizando-se do tipo de arquivo *Portable Pixmap* (PPM) é possível criar um texto em ASCII que é automaticamente convertido para uma imagem ao ser introduzido em um editor de imagem como GIMP ou Photoshop. O tipo PPM é composto da seguinte forma:

Figura 1: Representação em código ASCII de uma imagem colorida

```
1 P3
2 3 2
3 255
4 255 0 0
5 0 255 0
6 0 0 255
7 255 255 0
8 255 255 255
9 0 0 0
```

Fonte: Autoria própria.

Onde “P3” indica que está trabalhando com o sistema de cores aditivas “Red Green Blue” (RGB) no modelo ASCII, a segunda linha 3 2 indica o número de colunas e o número de linhas, respectivamente, já o terceira linha indica que se utiliza uma faixa de 0 até 255 dígitos números para representar as cores se utilizando de RGB. O sistema de cores RGB (Red, Green, Blue – Vermelho, Verde e Azul) é baseado na combinação aditiva de luz, onde diferentes intensidades desses três canais produzem todas as cores visíveis em telas digitais. Cada canal pode assumir valores de 0 a 255, permitindo mais de 16 milhões de cores distintas. Quando os três canais estão em zero (RGB(0,0,0)), o resultado é preto, pois não há luz; quando estão no máximo (RGB(255,255,255)), o resultado é branco, pois toda a luz está presente. Cores como vermelho (255,0,0), verde (0,255,0) e azul (0,0,255) são chamadas de cores primárias aditivas, e suas combinações produzem cores secundárias como amarelo (255,255,0), ciano (0,255,255) e magenta (255,0,255). Esse modelo é amplamente usado em monitores, TVs, câmeras digitais e na web, sendo essencial para a representação e

manipulação de imagens em ambientes digitais. As linhas sucessivas representam a composição de cada pixel em formato RGB. O resultado final é apresentado na Figura 2.

Figura 2: Exemplo de imagem colorida no sistema de cores RGB.



Fonte: Autoria própria.

O processo de renderização implementou o modelo de iluminação de Phong, que combina componentes difusos, especulares e ambientais para gerar imagens realistas. Para reduzir artefatos de aliasing, foi incorporada uma técnica de superamostragem que calcula múltiplos raios por pixel. O sistema de filtros para daltonismo foi integrado como estágio final do pipeline, aplicando as transformações de cor diretamente no buffer de pixels antes da geração da imagem final. A interface de linha de comando foi desenvolvida para oferecer flexibilidade na configuração dos parâmetros de renderização e seleção dos filtros de acessibilidade.

A escala que define quais cores devem ser alteradas para simular daltonismo, como tritanopia, é baseada no espaço de cor perceptual “*Long Medium Short*” (LMS), que corresponde às respostas dos três tipos de cones presentes na retina humana: L (sensível ao vermelho), M (sensível ao verde) e S (sensível ao azul). Para realizar a simulação, a imagem RGB é convertida para o espaço LMS utilizando uma matriz de transformação perceptual. Em seguida, é aplicada uma matriz específica que simula a deficiência visual, como a perda ou alteração na resposta dos cones S no caso da tritanopia. Após esse ajuste, a imagem é convertida novamente de LMS para RGB, permitindo visualizar o resultado na tela.

Além disso, todos os experimentos foram realizados em uma mesma máquina, equipada com processador Intel Core i5-10210U e 24 GB de memória RAM, a fim de garantir condições uniformes de execução. Essa padronização busca eliminar variáveis externas de desempenho, como diferenças de hardware, que poderiam interferir na avaliação comparativa entre os experimentos realizados.

3. RESULTADOS E DISCUSSÃO

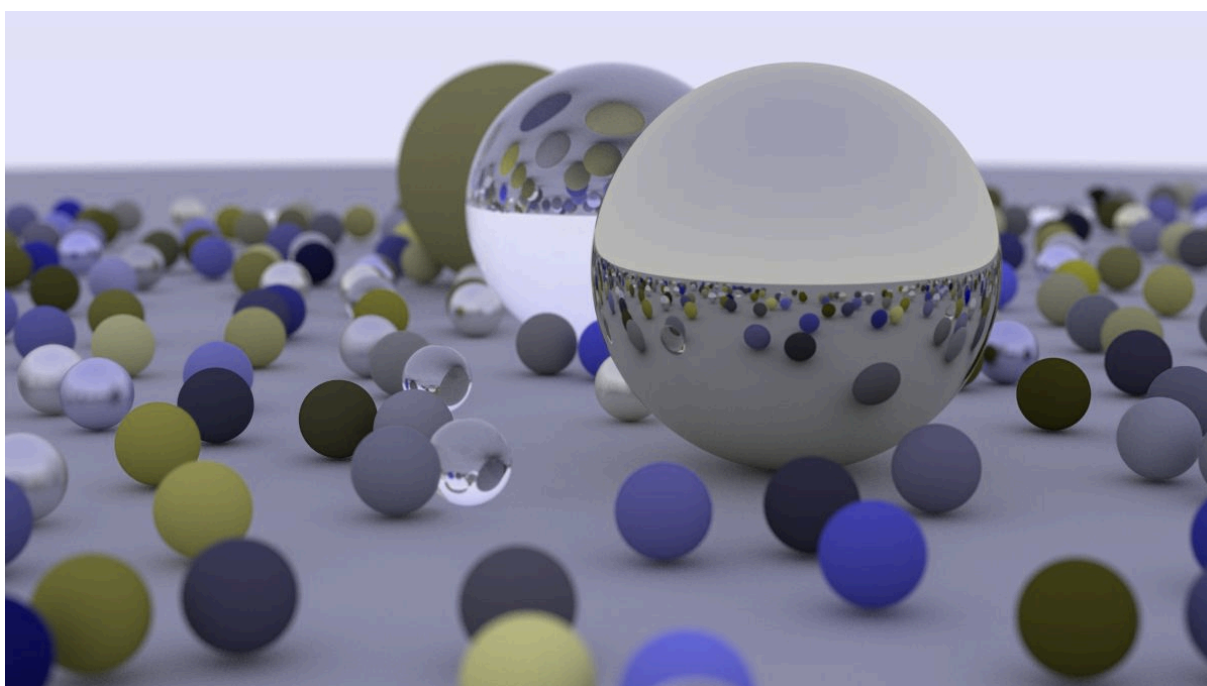
As Figuras 3, 4, 5 e 6 apresentam ilustrações de uma imagem gerada ao final do projeto sem a aplicação dos filtros, e com os filtros para Protanopia, para Deuteranopia e para Tritanopia, respectivamente.

Figura 3: Imagem gerada ao final do projeto, sem filtros



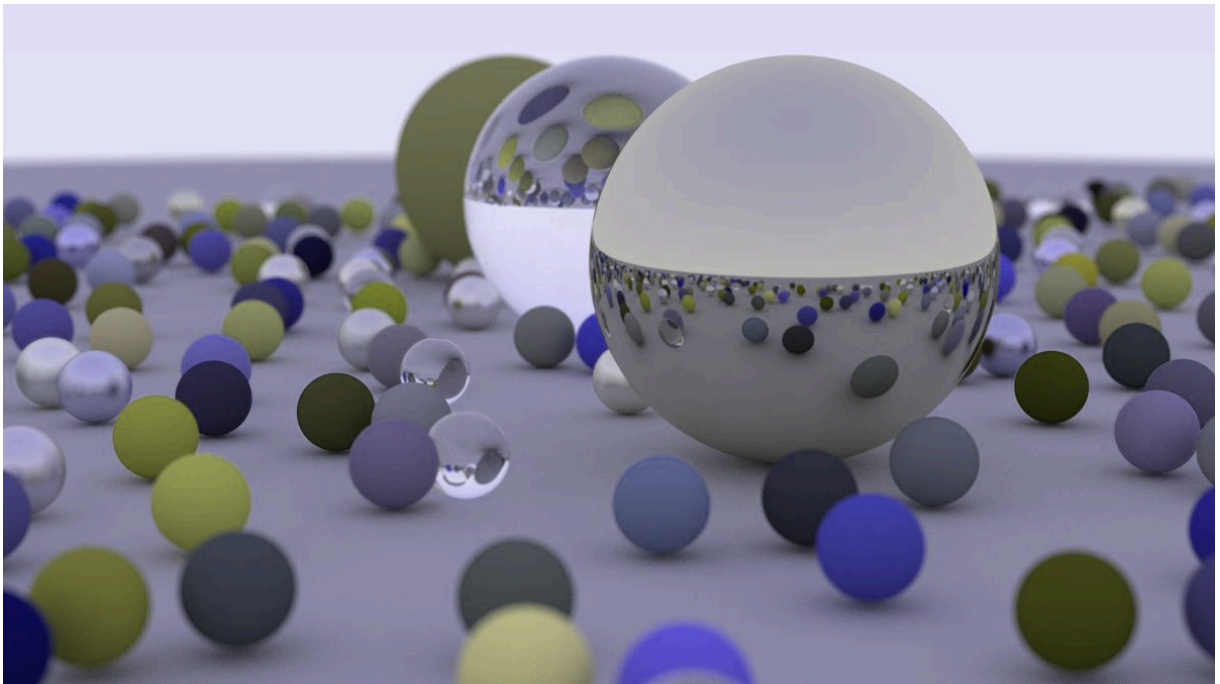
Fonte: Autoria própria.

Figura 4: Imagem gerada com filtro para Protanopia



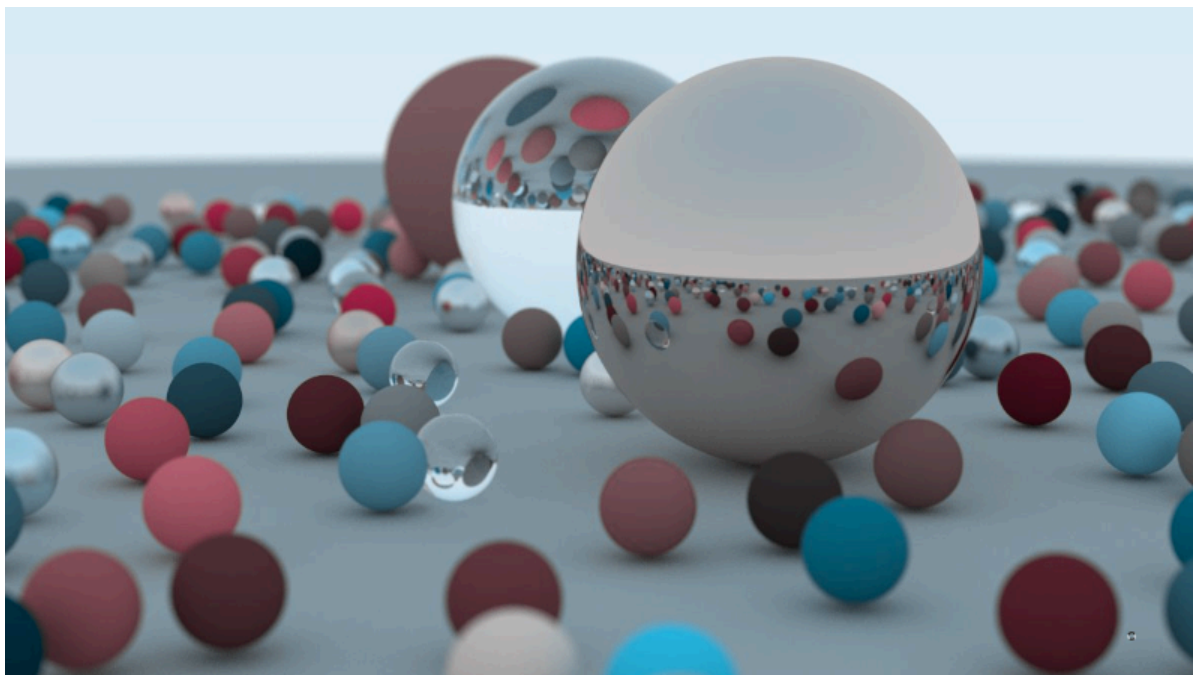
Fonte: Autoria própria.

Figura 5: Imagem gerada com filtro para Deuteranopia



Fonte: A autoria própria.

Figura 6: Imagem gerada com filtro para Tritanopia



Fonte: A autoria própria.

Os resultados obtidos estão consolidados na Tabela 1, a qual apresenta os diferentes cenários de teste avaliados e os resultados obtidos, em termos do tempo médio para renderização, em segundos (s), bem como o desvio-padrão entre os experimentos e a

sobrecarga computacional observada. Foram avaliados os cenários de renderização “sem filtro”, e os filtros para Protanopia, para Deuteranopia e para Tritanopia.

Os testes de desempenho revelaram que a implementação proporcionou uma cena de referência com três esferas e dois planos foi renderizada em média em 2,45 segundos com desvio padrão de 0,15 segundos. A aplicação dos filtros para daltonismo aumentou o tempo médio para 2,68 segundos, representando uma sobrecarga computacional de aproximadamente 9,4%. Esse resultado demonstra a viabilidade da abordagem mesmo sem o uso de aceleradores gráficos dedicados.

Tabela 1: Resultados obtidos nos experimentos realizados em diferentes cenários de uso.

| Cenário de Teste | Tempo Médio (s) | Desvio Padrão (s) | Sobrecarga Computacional |
|--------------------------|-----------------|-------------------|--------------------------|
| Renderização sem filtros | 2,45 | 0,15 | - |
| Protanopia | 2,61 | 0,17 | 6,5% |
| Deuteranopia | 3,00 | 0,16 | 7% |
| Tritanopia | 2,80 | 0,16 | 6,3% |

Fonte: Autoria própria.

A avaliação qualitativa dos filtros desenvolvidos foi realizada através de uma análise comparativa detalhada com sistemas de acessibilidade para daltonismo já implementados em motores de jogo e títulos comerciais consagrados. O estudo focou em examinar a eficácia das soluções propostas em relação às referências do mercado, utilizando como parâmetros principais a fidelidade na representação de cores e a manutenção da legibilidade dos elementos visuais.

Para esta análise, foram selecionadas as implementações de acessibilidade do Unity (UNITY, 2022), que oferece um pacote específico para simulação de daltonismo, e do Unreal Engine 5 (EPIC GAMES, 2023), com seu sistema nativo de suporte a deficiências de visão de cores. Além destes motores, foram incluídos na comparação jogos comerciais reconhecidos por seus avanços em acessibilidade, como World of Warcraft, Overwatch 2 e Final Fantasy XIV, todos com sistemas dedicados de filtros para daltonismo (BLIZZARD, 2022)(SQUARE ENIX, 2023).

Um aspecto importante observado durante os testes foi a influência direta do contraste original da cena na eficácia dos filtros. Cenas com iluminação bem balanceada e alto

contraste natural apresentaram os melhores resultados, enquanto ambientes com texturas muito detalhadas ou variações sutis de cor representaram desafios maiores para todos os sistemas testados. Elementos de interface e indicadores visuais críticos, como barras de vida e marcadores de objetivos, foram os que mais se beneficiaram da aplicação dos filtros em todos os cenários avaliados.

A análise revelou ainda que cenas com paletas de cores limitadas tendem a responder melhor aos filtros, provavelmente devido à menor complexidade na gama cromática original. Por outro lado, situações com grande variedade de tons e saturações mostraram maior variação nos resultados entre os diferentes sistemas testados.

Esta abordagem metodológica permitiu uma avaliação abrangente e técnica da eficácia dos filtros desenvolvidos, utilizando como base de comparação soluções já consolidadas e amplamente testadas no mercado. Os resultados obtidos fornecem indicativos consistentes sobre o desempenho do sistema proposto em relação ao estado da arte atual em acessibilidade visual para jogos e aplicações gráficas.

4. CONSIDERAÇÕES FINAIS

A pesquisa demonstrou que é possível implementar soluções eficientes para acessibilidade visual diretamente no pipeline de renderização, sem comprometer significativamente o desempenho do sistema. Os resultados sugerem que a abordagem baseada em transformações matriciais de cor representa uma solução robusta para auxiliar portadores de daltonismo em aplicações gráficas. Futuros trabalhos poderão explorar a aceleração em GPU desses algoritmos e a integração com técnicas mais avançadas de pós-processamento. A metodologia desenvolvida neste trabalho estabelece um framework replicável para inclusão de recursos de acessibilidade em motores gráficos educacionais e profissionais.

REFERÊNCIAS

AKENINE-MÖLLER, T.; HAINES, E.; HOFFMAN, N. **Real-Time Rendering**. 4th ed. Boca Raton: CRC Press, 2018.

BLIZZARD. **Accessibility in Overwatch 2**. Irvine: Blizzard Technical Blog, 2022. Disponível em: <https://playoverwatch.com>. Acesso em: 10 Junho 2025.

DIRECT3D 11 GRAPHICS. Microsoft. Disponível em: <https://docs.microsoft.com/en-us/windows/win32/direct3d11/>. Acesso em: 21 maio 2025.

EPIC GAMES. **Color Vision Deficiency Tools in UE5**. Raleigh: Unreal Engine Documentation, 2023. Disponível em: <https://docs.unrealengine.com>. Acesso em: 10 Junho 2025.

FERNANDO, R. (Ed.). **GPU Gems: Programming Techniques, Tips and Tricks for Real-Time Graphics**. Boston: Addison-Wesley, 2004.

NAPOLI, Daniela; CHIASSON, Sonia. **Exploring the Impact of Colour-Blindness on Computer Game Performance**. In: CHI Conference on Human Factors in Computing Systems - Extended Abstracts, 2018, Montreal. Disponível em: <https://chorus.scs.carleton.ca/wp-content/papercite-data/pdf/napoli2018-colourblindgame-chilbw.pdf>. Acesso em: 30 maio 2025.

SHIRLEY, P.; MARSCHNER, S. **Fundamentals of Computer Graphics**. 4th ed. Boca Raton: CRC Press, 2020.

SQUARE ENIX. **Final Fantasy XIV: Dawn Trail Patch Notes**. Tóquio: Square Enix, 2023. Seção 7.0. Disponível em: <https://na.finalfantasyxiv.com>. Acesso em: 30 maio 2025.

UNITY. **Color Blindness Simulator Asset**. San Francisco: Unity Asset Store, 2022. Disponível em: <https://assetstore.unity.com>. Acesso em: 29 Maio 2025.

ZHU, Zhenyang; MAO, Xiaoyang. Image recoloring for color vision deficiency compensation: a survey. **The Visual Computer**, [S.l.], v. 37, n. 12, p. 2999–3018, 2021. Springer. DOI: <https://doi.org/10.1007/s00371-021-02240-0>

BRETTEL, H.; VIÉNOT, F.; MOLLON, J. D. **Computerized simulation of color appearance for dichromats**. *JOSA A*, v. 14, n. 10, p. 2647-2655, 1997. DOI: 10.1364/JOSAA.14.002647