

O Custo do Fotorrealismo: Uma Análise Comparativa de Pipelines de Renderização e o Papel da Inteligência Artificial em Hardware de Entrada

Vitória Elizabeth F. Silva¹, Iago Medeiros¹

¹Faculdade de Engenharia de Computação – Universidade Federal do Pará (UFPA)
Tucuruí – PA – Brasil

vitoria.silva@tucuruui.ufpa.br, iagomedeiros@ufpa.br

Abstract. *With the evolution of real-time computer graphics, games are increasingly being marked by the transition from the traditional rasterization technique to advanced techniques, such as Ray Tracing and Path Tracing, which bring an elevated and unprecedented level of photorealism. However, due to their high computational cost, these advanced rendering techniques present a limitation for entry-level hardware, which constitutes a large part of the Brazilian market. This work presents a quantitative and qualitative analysis comparing the different types of pipelines, using an NVIDIA GeForce RTX 3050 GPU for the case study, combined with NVIDIA DLSS (Deep Learning Super Sampling) as an enabling component. The results showed that although DLSS enables a satisfactory experience during gameplay with Ray Tracing, the same does not occur with Path Tracing. Even with DLSS, the generated frame rate remained below the 30 FPS acceptable for fluid gameplay, which compromises playability, making Path Tracing unfeasible on this class of hardware.*

Resumo. *Com a evolução da computação gráfica em tempo real, cada vez mais os jogos estão sendo marcados pela transição da técnica tradicional de rasterização para técnicas avançadas como Ray Tracing e Path Tracing, que trazem um fotorrealismo elevado e sem precedentes. Porém, devido seu elevado custo computacional, estas técnicas de renderização avançadas apresentam uma limitação para os hardwares de entrada, que constitui uma grande parte do mercado brasileiro. Este trabalho apresenta uma análise quantitativa e qualitativa comparando os diferentes tipos de pipelines, utilizando uma GPU NVIDIA GeForce RTX 3050 para o estudo de caso, aliada a tecnologia de super-resolução NVIDIA DLSS (Deep Learning Super Sampling) como componente viabilizador. Os resultados mostraram que embora o DLSS viabilize uma experiência satisfatória durante a gameplay com Ray Tracing, o mesmo não se verifica com Path Tracing, que mesmo utilizando DLSS, a taxa de quadros gerada ficou abaixo dos 30 FPS (Frames Per Seconds) aceitáveis para uma gameplay fluída, o que compromete a jogabilidade, tornando o Path Tracing inviável nesta classe de hardware.*

1. Introdução

A computação gráfica em tempo real se iniciou com a rasterização, que se consolidou como a técnica dominante entre os jogos até os dias atuais, é o pipeline mais eficiente, que

se baseia na conversão de objetos em 3D para *pixels* em 2D [Corporation 2018]. Apesar de ser uma técnica eficiente e que alcançou resultados visuais satisfatórios, ela possui limitações, resultando em imagens menos realistas e com baixa fidelidade se comparada à imagens reais.

Através do advento de *hardware* com aceleração dedicada (GPUs dedicadas), surgiu um novo paradigma, o *Ray Tracing* em tempo real com *Battlefield V* (2018, desenvolvido pela DICE e publicado pela *Electronic Arts*), sendo este um dos primeiros jogos a trazer esta tecnologia de forma oficial. Logo em seguida, em 2020, o jogo *Cyberpunk 2077* (desenvolvido e publicado pela *CD Projekt Red*) trouxe oficialmente o *Path Tracing*, sendo uma versão ainda mais completa e realista em comparação ao seu antecessor. [Kelling et al. 2024] destacam que este avanço demanda métodos como LiPac (*Light Path Guided Culling for Hybrid Real-Time Path Tracing*), técnica usada para analisar previamente quais raios contribuem mais para uma entrega fiel a de uma iluminação real cortando os caminhos pouco relevantes para acelerar o *Path Tracing* híbrido, enquanto [Mohanto et al. 2024] exploram *foveated rendering*, uma técnica baseada em algoritmos que simulam *eye-tracking* para detectar a direção na qual o jogador está olhando com intuito de melhorar a qualidade visual naquela região em específico, sendo uma alternativa perceptível para reduzir o custo computacional.

Ambas as técnicas trabalham através da simulação do caminho dos raios de luz, calculando reflexos, sombras e refrações, mas enquanto o *Ray Tracing* simula o caminho de um raio de luz [Corporation 2018], o *Path Tracing* simula o caminho completo da luz [Neri 2022], sendo extremamente custoso principalmente para hardwares de entrada. Essa dificuldade foi observada também por [Kelling 2024] ao ter avaliado a integração de um Path Tracer eficiente em memória voltado a jogos e concluiu que, apesar das otimizações, esta técnica permanece inviável

O *Ray Tracing* é exigente quando se trata de processamento de imagem, se tornando quase impraticável se não estiver atrelado a uma tecnologia de super-resolução por IA (Inteligência Artificial). A partir dessas limitações, o DLSS (*Deep Learning Super Sampling*), lançado em 2018, surgiu como a principal resolução para o problema de desempenho. Ao renderizar uma imagem numa resolução inferior e utilizar redes neurais para reconstruir a imagem para a resolução desejada (por exemplo, de 1080p para 2K) [Corporation nd], o DLSS veio como promessa de atenuação o custo do *Ray Tracing*, se tornando mais acessível a um público maior, consolidando-se hoje um componente fundamental para a renderização de jogos atuais. Segundo a [Corporation 2025a], a versão mais recente, DLSS 4, introduziu geração *multi-frame* e métodos de reconstrução, consolidando a IA como viabilizadora do fotorrealismo em tempo real.

Este trabalho, investiga a eficácia desta solução num cenário relevante: a experiência em hardware de entrada, entendido aqui como dispositivo de menor custo e desempenho limitado em comparação a configurações medianas e de alto desempenho. O objetivo é realizar uma análise aprofundada dos pipelines de Rasterização, *Ray Tracing* e *Path Tracing*, utilizando uma *GPU NVIDIA GeForce RTX 3050* como estudo de caso. O restante do trabalho está organizado da seguinte forma: Seção 2 compila os principais trabalhos relacionados com a pesquisa. Seção 3 apresenta a análise comparativa de pipelines de renderização. Seção 4 é a avaliação comparativa. Seção 5 conclui a pesquisa reunindo as principais discussões acerca do papel da I.A. em hardware de entrada.

2. Trabalhos Relacionados

Com avanços crescentes em técnicas de renderização em tempo real, várias pesquisas têm buscado harmonizar a qualidade gráfica e desempenho no contexto de jogos digitais. O trabalho de [Vries 2023] implementa *Ray Tracing* no motor de jogo *Unity* com o objetivo de melhorar a qualidade gráfica, explicitando melhorias visuais expressivas, mas evidenciando o impacto no desempenho. Apesar de prático, esse estudo limitou-se apenas a um único motor gráfico e não considerou recursos de *Upscaling*, que é o processo de converter uma imagem de resolução mais baixa para uma resolução mais alta através de algoritmos tradicionais ou por IA [NVIDIA 2020], que poderiam reduzir o impacto sobre o hardware.

No trabalho de [Wen et al. 2025] foi explorada a super-resolução em tempo real, com foco em soluções mais leves como alternativa a técnicas proprietárias, aproximando da qualidade do DLSS sem necessitar de hardware específico. No entanto, embora os resultados sejam promissores, o estudo restringiu-se apenas ao impacto da super-resolução sem avaliar seu impacto entre os diferentes paradigmas de renderização presentes nos jogos.

O trabalho de [Tan et al. 2022] apresentou uma abordagem híbrida para a técnica visual *motion blur*, um tipo de efeito visual que causa borramento quando objetos ou a câmera do jogo se movimentam durante a geração de imagem [StudioBinder 2021], combinando rasterização e *Ray Tracing* para apresentar resultados mais realistas em jogos. Essa proposta mostrou-se eficiente, mas permanece restrita a um efeito visual específico focado na estética do jogo, não avançando para uma análise mais ampla sobre a viabilidade de diferentes pipelines de renderização no cenário atual.

Todos os trabalhos discutidos contribuem para a evolução da renderização em tempo real, porém, apresentam lapsos na literatura pois se concentram em motores gráficos ou efeitos visuais específicos e não discute a relação entre os diferentes paradigmas de renderização e as técnicas modernas de super-resolução e também de *anti-aliasing*, uma técnica para suavizar bordas e atenuar serrilhados tornando as imagens realistas, baseadas em IA [Lenovo 2024]. Neste contexto, a contribuição principal deste trabalho está em oferecer uma análise comparativa entre rasterização, *Ray Tracing* e *Path Tracing*, salientando o papel das tecnologias de *upscaling* na viabilidade de gráficos de alta qualidade nos jogos, balanceando realismo e custo computacional.

3. O Custo do Fotorrealismo: Uma Análise Comparativa de Pipelines de Renderização e o Papel da Inteligência Artificial em Hardware de Entrada

Foi definido um procedimento experimental. Esta seção detalha o ambiente de teste, os objetos de estudo selecionados, o procedimento da coleta de dados e as métricas utilizadas para a análise comparativa entre os três *pipelines* de renderização.

O DLSS é uma tecnologia distribuída pela NVIDIA, que utiliza IA para renderizar o jogo através de algoritmos de aprendizado profundo e dados de *frames* prece-dentes numa resolução interna inferior para que o desempenho seja aumentado. Após a análise de IA, o DLSS reconstrói a imagem para uma resolução mais alta, seguindo-se de um resultado análogo ao que seria alcançado na renderização nativa [Corporation nd]. [Wen et al. 2025] exploram esse caminho ao propor uma técnica de super-resolução leve para gráficos, reforçando a importância desta análise conduzida neste trabalho.

Por outro lado, o DLAA (*Deep Learning Anti-Aliasing*) também é distribuído pela NVIDIA, utiliza Inteligência Artificial para atenuar serrilhados nos jogos e possui uma técnica parecida com o DLSS, aumentando o desempenho do jogo renderizando a uma resolução inferior mas sem *upscaling* [Corporation 2025b]. Esta tecnologia aponta para a melhor qualidade visual possível. O trabalho de [Hu et al. 2023] foca numa abordagem semelhante de *anti-aliasing* multiescala, baseadas em redes neurais, reforçando a necessidade de uma integração em diferentes etapas do pipeline de renderização.

3.1. Objetos de Estudo

Foram selecionados 8 jogos para realizar as análises das pipelines, eles estão categorizados de acordo com as tecnologias de renderização suportadas para permitir uma análise multifacetada conforme a Tabela 1.

Jogo	Tecnologias Suportadas	Configurações Testadas (Full HD)
<i>Cyberpunk 2077</i>	Rasterização, Ray Tracing, Path Tracing, DLSS	Ultra: Rasterização (Nativo e DLSS Automático), Ray Tracing (Nativo e DLSS Automático), Path Tracing (DLSS Automático).
<i>Alan Wake 2</i>	Rasterização, Ray Tracing, Path Tracing, DLSS, DLAA	Baixo: Rasterização, Ray Tracing e Path Tracing (DLSS Automático). Testes adicionais em HD com DLAA.
<i>Control</i>	Rasterização, Ray Tracing, DLSS	Alto: Rasterização (Nativo e DLSS Automático), Ray Tracing (Nativo e DLSS Automático).
<i>Guardians of the Galaxy</i>	Rasterização, Ray Tracing, DLSS	Alto: Rasterização (Nativo e DLSS Ultra Desempenho), Ray Tracing (Nativo e DLSS Ultra Desempenho).
<i>Batman: Arkham Knight</i>	Apenas Rasterização	Máximo: Rasterização.
<i>Dead Space 2</i>	Apenas Rasterização	Máximo: Rasterização.
<i>Portal</i> (2007)	Apenas Rasterização	Alto: Rasterização (Nativo).
<i>Portal with RTX</i>	Rasterização, Path Tracing, DLSS	Alto: Rasterização (Nativo), Path Tracing (DLSS Automático).

Tabela 1. Tecnologias e configurações testadas para os jogos analisados.

- Análise Completa (Rasterização, Ray Tracing Completo, Path Tracing):** *Cyberpunk 2077* (CD Projekt Red, 2020) e *Alan Wake 2* (Remedy Entertainment,

2023). Estes jogos serviram como base para a comparação direta entre os três principais *pipelines*.

2. **Análise de Ray Tracing Completo:** *Control* (Remedy Entertainment, 2019) e *Marvel's Guardians of the Galaxy* (Eidos-Montréal, 2021). Utilizados para um estudo de caso aprofundado sobre implementações de *Ray Tracing* Completo e o impacto dele nos efeitos específicos.
3. **Rasterização Pura:** *Batman: Arkham Knight* (Rocksteady Studios, 2015) e *Dead Space 2* (Visceral Games, 2011). Selecionados como pontos de referência de alta qualidade visual alcançada apenas com técnicas de rasterização, considerada a técnica mais primitiva e menos realista entre as demais.
4. **Path Tracing Puro:** *Portal* (Valve, 2007) e *Portal with RTX* (NVIDIA, 2022). Usado apenas para a análise qualitativa mais impactante, apresentando a mudança visual completa proporcionada pelo *Path Tracing*.

3.2. Ambiente Experimental

Todos os testes foram realizados num único *hardware* com a finalidade de garantir a consistência dos resultados, onde estavam em execução apenas os jogos em modo *offline* e o software de coleta de dados. As especificações do ambiente de testes estão presentes na Tabela 2.

Componente	Especificação
CPU	Intel Core i5-13450HX
GPU	NVIDIA GeForce RTX 3050 6GB (Laptop)
RAM	16 GB DDR5 4800MHz
Armazenamento	1 TB NVMe SSD
Sistema Operacional	Windows 11 versão 24H2
Driver Gráfico	NVIDIA Game Ready Driver 576.88
Resolução	1920 x 1080 (Full HD)

Tabela 2. Especificações do Ambiente Experimental.

3.3. Procedimento de Coleta de Dados

Para cada configuração gráfica testada em cada jogo, foi seguido um padrão para a realização dos testes. Esse procedimento foi feito para garantir a consistência dos dados a serem analisados:

3.3.1. Procedimentos para a Análise Quantitativa:

- **Duração e Repetição:** Durante a *gameplay*, o percurso foi efetuado por três minutos. Foi definido o mesmo percurso para todas as configurações. Cada configuração teve o seu teste feito seis vezes. Os resultados das seis execuções foram usados para calcular a média dos dados e o desvio padrão, garantindo dados mais precisos.
- **Coleta de Logs:** Os dados das execuções foram coletados utilizando a ferramenta de *log* do software *MSI Afterburner*, dados estes que capturam a variação das métricas ao longo do tempo.

- **Captura de Imagens:** Para a análise visual em jogo, foram realizadas capturas em formato `.png`, que não possui perdas de compressão. Para garantir uma comparação de imagem justa, todas as capturas foram tiradas no mesmo ponto de vista, ângulo e posicionamento dos personagens, alterando apenas as configurações gráficas e utilizando o modo foto dos jogos que possuem essa funcionalidade.

3.3.2. Procedimentos para a Análise Visual Qualitativa:

Foi feita uma análise das capturas posicionadas lado a lado, focando em elementos técnicos como a qualidade de reflexos, a precisão das sombras, a naturalidade da iluminação global e a oclusão de ambiente. Para esta análise comparativa visual, foi utilizada a ferramenta NVIDIA ICAT (*Image Comparison & Analysis Tool*). Esta ferramenta é conhecida por garantir que as imagens estejam na mesma cena, com o alinhamento exato, com configurações de imagens diferentes e permitindo uma análise visual correta e justa. As capturas de tela efetuadas nesta análise estão disponíveis para consulta online¹.

4. Resultados e Análise

Nesta seção, serão apresentados os gráficos obtidos através dos registros de *log* e *benchmark* realizados com o MSI *Afterburner* e analisado os dados qualitativos e quantitativos obtidos através do processo que foi descrito na seção anterior. A estrutura da análise está obedecendo uma base de comparação entre rasterização, em seguida partindo para a análise de custo dos pipelines mais avançados e o papel da tecnologia DLSS no desempenho das diferentes técnicas de renderização, e finalizando na análise da qualidade visual.

4.1. Métricas Avaliadas

Para este trabalho foram analisadas métricas padrão da indústria de jogos com o objetivo de analisar o desempenho e fluidez em cada um dos pipelines de renderização apresentados.

FPS (*Frames per Second*): É a quantidade de quadros (médio) que estão sendo renderizados por segundo. Essa métrica varia a cada *frame*, mostrando o desempenho em tempo real. Sua fórmula está representada na Equação 1:

$$FPS = \frac{1}{T_{\text{quadro}} (s)} \quad (1)$$

Tempo de Quadro: Essa métrica apresenta o tempo no qual o *hardware* leva para renderizar um único quadro. Quanto menor o tempo de quadro, mais rápido e mais fluido é o jogo. Sua fórmula está representada na Equação 2:

$$\bar{T}_{\text{quadro}} = \frac{1}{N} \sum_{i=1}^N T_{\text{quadro},i} \quad (2)$$

¹<https://encurtador.com.br/hdqBP>

FPS 1% Low: Indica o FPS correspondente ao pior 1% dos *frames*, ou seja, o pior momento de desempenho do jogo. Quanto menor for o valor do 1% Low, mais perceptível é a queda de desempenho. Sua fórmula está representada na Equação 3:

$$FPS_{1\% \text{ low}} = \frac{1000}{T_{99p} \text{ (ms)}} \quad (3)$$

4.2. Análise da *Baseline* de Rasterização

Com o objetivo de contextualizar o desempenho das tecnologias mais avançadas, primeiramente foi estabelecida uma *baseline* focada em performance. Essa comparação foi preparada a partir dos modos de rasterização otimizada entre jogos diversos em configurações adaptadas para o *hardware* de teste, globalizando motores gráficos distintos e gerações tecnológicas. A Figura 1 apresenta o comparativo de desempenho entre estas configurações avaliando o FPS médio dos jogos pelo tempo de teste (três minutos).

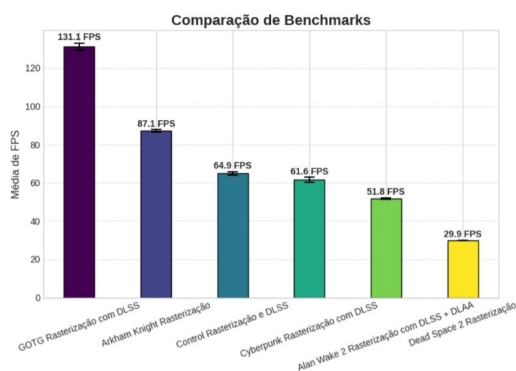


Figura 1. Comparativo de desempenho de baseline de rasterização

Foi observado que *Marvel's Guardians of the Galaxy*, atingiu uma robusta média de 131.1 FPS e desvio padrão de 1.93 FPS na configuração mais alta (Ultra) com DLSS. Ainda que seja um jogo relativamente recente, que tendem a ser mais pesados, este é um resultado extremamente satisfatório dada a qualidade visual deste jogo. Em seguida foi observado que o motor gráfico de *Batman: Arkham Knight*, operando em seu único modo de renderização alcançou uma média de 87.1 FPS e desvio padrão de 0.63 FPS na configuração mais alta (Alta), que apesar de seu método de renderização mais antigo, entrega resultados visuais impressionantes considerando a tecnologia existente para a época.

Posteriormente temos o jogo *Control* com uma média de 64.9 FPS e desvio padrão de 0.82 FPS na configuração Alta com DLSS automático, seguidamente de *Cyberpunk 2077* com uma média de 61.6 e desvio padrão de 1.29 FPS na configuração mais alta (Ultra) com DLSS.

Após, foi analisado que o jogo mais pesado e mais atual da lista, *Alan Wake 2* obteve uma média de 51.8 FPS e desvio padrão de 0.33 FPS na configuração Baixo com DLSS automático e DLAA (1920x1080). Por fim, o último e mais antigo da lista, *Dead Space 2* com a média de 29.9 FPS Na configuração mais alta (*Very High*) e se mantendo como o mais estável de todos com o desvio padrão de 0.0 FPS. Este valor baixo de FPS

médio se deve pela limitação própria do jogo que fixa o valor do FPS em 30, não sendo possível remover esta limitação. Apesar de ser um jogo antigo, *Dead Space 2* entrega excelentes visuais até para a época atual.

4.3. Análise Comparativa do Desempenho dos Pipelines

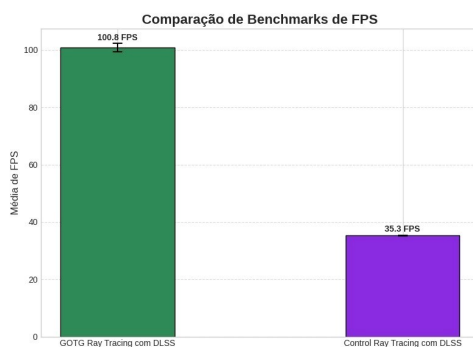


Figura 2. An lise de desempenho dos jogos que possuem *Ray Tracing* nativo

Esta subse  o apresenta uma vis o geral dos pipelines com os dois jogos que possuem suporte nativo   Rasteriza  o, *Ray Tracing* e *Path Tracing*. O objetivo desta compara  o   mostrar de forma clara e direta o custo de desempenho a ser pago ao selecionar a tecnologia de renderiza  o desejada, desde a tecnologia mais leve (Rasteriza  o) at  a mais pesada (*Path Tracing*), com o aux lio do *DLSS*.

Na Figura 2   apresentado um gr fico de desempenho de *Ray Tracing* nos jogos *Marvel's Guardians of the Galaxy* e *Control* aliados ao *DLSS*. No jogo *Marvel's Guardians of the Galaxy* foi atingida uma m dia de 100.8 FPS, um valor elevado que garante uma jogabilidade extremamente fluida. O desvio padr o de 1.46 FPS, que indicou instabilidade, mas sem grandes intercorr ncias.

Para o jogo *Control* foi registrado uma m dia de 35.5 FPS, sendo considerado um desempenho bem mais baixo do que o jogo anterior. Por m, se mostrou mais est vel com um desvio padr o de 0.15 FPS. Apesar de ser desempenho baixo, n o variou significativamente durante a sua execu  o.

Na Figura 3 temos o gr fico que apresenta o desempenho dos pipelines no jogo *Cyberpunk 2077* em conjunto com *DLSS*. Na configura  o de Rasteriza  o com *DLSS*, o jogo entrega o maior desempenho com m dia de 61.6 FPS e desvio padr o de 1.3 FPS.

Na configura  o de *Ray Tracing* com *DLSS*, o FPS m dio cai quase pela metade em rela  o   rasteriza  o, atingindo a m dia de 31.4 FPS e um desvio padr o de 4.96 FPS, o que indica uma grande instabilidade, apontando uma queda de 49% do desempenho em rela  o a configura  o anterior. Este fato mostra o peso da ilumina  o em tempo real. Por fim, o *Path Tracing* foi o mais pesado, caindo para 12.1 FPS e um desvio padr o de 0.5 FPS, mostrando uma queda de desempenho de 80% em rela  o   rasteriza  o com

DLSS. Isso torna este modo de renderização injogável, comprometendo a jogabilidade minimamente aceitável de 30 FPS [Spjut et al. 2022].

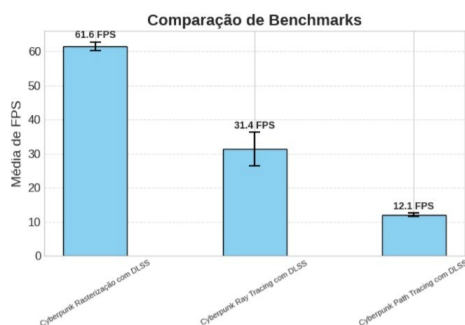


Figura 3. Comparação de desempenho dos Pipelines no jogo *Cyberpunk 2077*

Na Figura 4 é apresentada a análise do tempo de quadro para o jogo *Cyberpunk 2077*, percebeu-se que a melhor configuração foi a rasterização com DLSS, com tempo médio de 16,25 ms. Para a rasterização sem DLSS, notou-se uma queda de desempenho de 43% em relação à configuração anterior, indo para 23.31 ms. Ao ativar *Ray Tracing* com DLSS, o tempo de quadro dobrou, mostrando-se 96% mais lento que rasterização com DLSS (31,88 ms). Para o *Ray Tracing* sem o apoio do DLSS, o desempenho cai expressivamente, aumentando para 289% em relação ao valor de referência, chegando a 63,29 ms. Finalizando o gráfico, *Path Tracing* com DLSS é o pior cenário, com um aumento de 407% comparado ao referencial, chegando aos 82,42 ms.

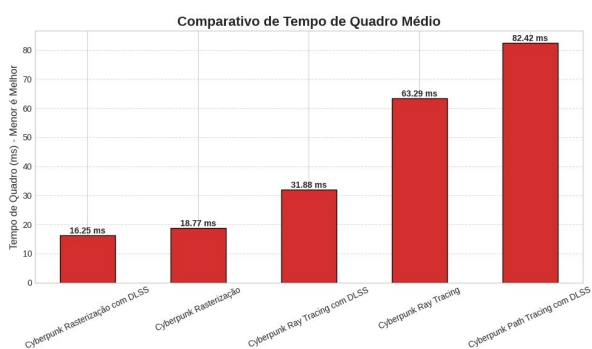


Figura 4. Comparativo de tempo de quadro no jogo *Cyberpunk 2077*.

Na Figura 5 é apresentado o desempenho dos pipelines no jogo *Alan Wake 2*. Após a análise, foi concluído que em termos de FPS é ainda mais pesado que o jogo anterior nas mesmas técnicas (mesmo utilizando DLSS automático). Na rasterização foi alcançada a média de 51.8 FPS e desvio padrão de 0.32 FPS, indicando estabilidade. Em relação a

rasterização, o *Ray Tracing* sofre uma queda de 70% do desempenho (15.5 FPS) com desvio padrão de 0.27 FPS. E o *Path Tracing* assume um nível impraticável, com 14.0 FPS e desvio padrão de 0.3 FPS, caindo para 73% em relação ao valor referencial.

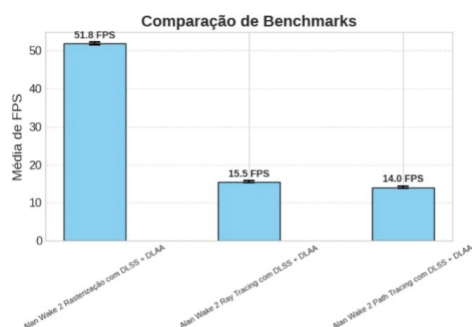


Figura 5. Comparação de desempenho dos Pipelines no jogo Alan Wake 2

Para o tempo de quadro analisado no jogo *Alan Wake 2* na Figura 6, o valor de referência será o da Rasterização com DLSS e DLAA, com 19,29 ms. Ao ativar o *Ray Tracing* com DLSS e DLAA, o tempo médio aumentou para 64.59 ms, um aumento 234% em relação à rasterização. O *Path Tracing* com DLSS e DLAA, degradou ainda mais a performance, alcançando os 71,26 ms, representando um aumento de 269% em relação a rasterização.

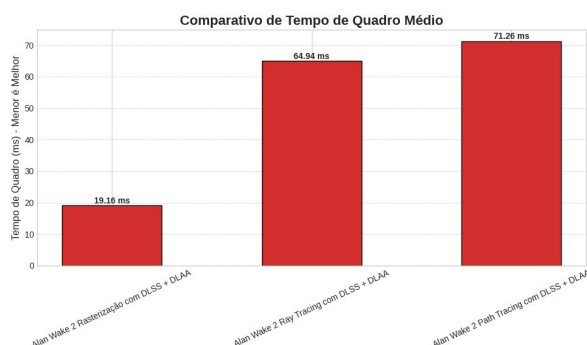


Figura 6. Comparativo de tempo de quadro no jogo Alan Wake 2.

Em ambos os jogos, o *Path Tracing* reduz drasticamente o desempenho, mesmo com o auxílio do DLSS, os ganhos na qualidade visual não compensam a perda de jogabilidade fluida. Este resultado é compatível com as conclusões de [Mohanto et al. 2024], que testaram o uso de *foveated rendering* para abrandar a carga do *Path Tracing*. Embora houvesse ganhos relevantes, a técnica não é o suficiente para viabilizar uma jogabilidade minimamente fluida em dispositivos de baixo custo.

4.4. Análise de Estabilidade

Para a análise de estabilidade, foram selecionados novamente os dois jogos que possuem os 3 *pipelines*. Conforme a Figura 7, a linha azul representa a rasterização, acima dos 60 FPS com algumas quedas. Esta configuração apresenta uma variação menor, ficando dentro do desvio padrão de 10 FPS da média, indicando boa estabilidade. O 1% *Low* atingiu 28 FPS, o que sugere uma rara queda brusca perceptível.

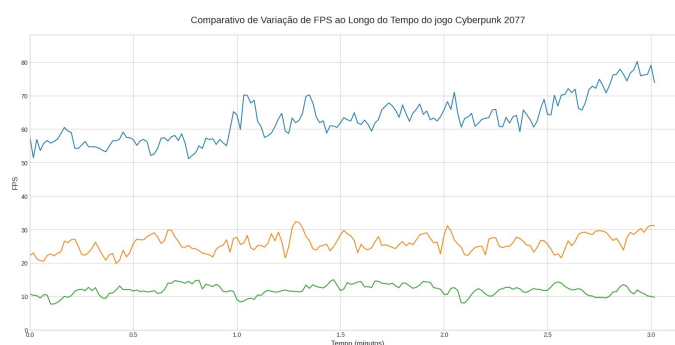


Figura 7. Análise de estabilidade no jogo *Cyberpunk 2077*

Na variação de estabilidade de FPS para o *Ray Tracing*, representado pela linha laranja, permaneceu na faixa de 25 FPS, com quedas bruscas em certos momentos. O *Ray Tracing* apresentou menor variação do que a configuração anterior. O 1% *Low* atingiu os 16 FPS, o que significa que há travadas perceptíveis com mais frequência.

Na última configuração do gráfico, o modo *Overdrive (Path Tracing)* representado pela linha verde se manteve abaixo dos 12 FPS e o 1% *Low* atingiu 7 FPS. As quedas abaixo de 10 FPS comprometem a *gameplay* e causam a sensação de travamento e lentidão. Apesar de relativamente estável, os valores são extremamente baixos.

Na Figura 8, temos o gráfico de estabilidade de *Alan Wake 2*. Nele é possível ver que a Rasterização com DLSS, representada pela linha azul, se manteve na faixa acima de 50 FPS, sendo a única configuração jogável. O 1% *Low* de 22 FPS indica quedas que podem afetar a fluidez, mas não tão drasticamente se comparado aos outros modos.

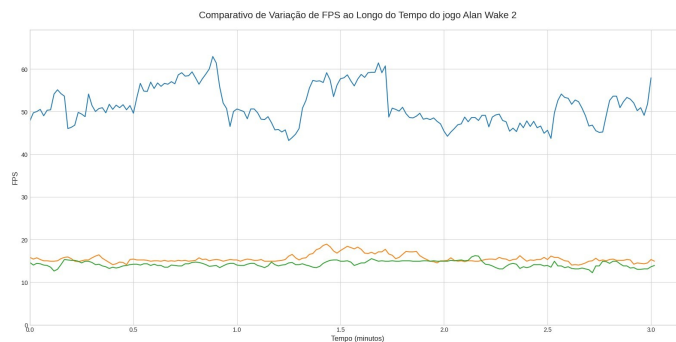


Figura 8. Análise de estabilidade no jogo Alan Wake 2

Em seguida temos o *Ray Tracing* combinado ao DLSS. Representado pela linha laranja, se manteve na média de 15 FPS e o 1% Low de 7 FPS, sugerindo problemas de *stutter* perceptíveis. Por fim, temos o *Path Tracing* com DLSS sendo representado pela linha verde. Extremamente baixa, na média de 14 FPS e 1%Low 6 FPS. Apesar de estável, está em um patamar injogável com um desempenho extremamente baixo. A estabilidade dos pipelines mostrou que, enquanto a rasterização apresenta uma variação controlada, o *Ray Tracing* sofre com mais oscilações, enquanto o *Path Tracing* atinge um nível de instabilidade crítica. [Tan et al. 2024] demonstraram que ao dividir a carga computacional com rasterização local e *Ray Tracing* delegado para a nuvem, é possível obter sombras mais realistas em tempo real mesmo em hardwares de entrada.

A análise dos três pipelines nos dois jogos mostrou que *Cyberpunk 2077* consegue entregar uma experiência satisfatória com a rasterização e um uso limitado de *Ray Tracing*. Mas para *Alan Wake 2*, apenas a rasterização mostrou um desempenho aceitável.

4.5. Comparação Visual



Figura 9. Comparação visual entre as 3 técnicas de renderização disponíveis em Cyberpunk 2077: Rasterização, Ray Tracing, Path Tracing

Na Figura 9, a imagem à esquerda representa a rasterização aliada ao DLSS, onde os reflexos do chão e nas superfícies causam a impressão de artificialidade. Ademais, a rasterização possui sombras e luzes menos naturais. Em relação a detalhes de superfície, a imagem apresentada parece mais lisa e menos detalhada. Na imagem do centro, temos *Ray Tracing* aliado ao DLSS, que apresenta reflexos de luz e do carro mais realista

com brilho melhor distribuído. No quesito iluminação, esta técnica traz uma iluminação mais realista se comparada à técnica anterior. Se tratando de detalhes de superfície, esta técnica apresenta mais materialidade. Por fim, na imagem à direita, temos a técnica mais avançada, o *Path Tracing*, que apresenta reflexos extremamente precisos, incluindo iluminação indireta, além de cores e luzes mais naturais. No quesito iluminação global, esta tecnologia captura a iluminação de ambiente de forma mais fiel à realidade. E se tratando de detalhes de superfície, o *Path Tracing* apresentou a melhor definição dos materiais apresentados na imagem, como neon e metal.

Na comparação qualitativa, o *Path Tracing* se destacou pela qualidade dos reflexos e iluminação, mas com um custo inviável para placas como a NVIDIA GeForce RTX 3050. Este contraste é abordado na pesquisa de [Li et al. 2024], que demonstram que abordagens neurais e super resolução reconstróem a iluminação de forma mais eficaz, se aproximando da qualidade do *Path Tracing* com um custo reduzido.

5. Conclusão

Neste trabalho, foi investigado o custo computacional do fotorrealismo em um hardware de entrada, um segmento expressivo no mercado brasileiro, por meio de uma análise quantitativa e qualitativa dos pipelines de renderização: Rasterização, *Ray Tracing* e *Path Tracing*. Foi utilizada uma GPU NVIDIA GeForce RTX 3050 como estudo de caso, a pesquisa avaliou o impacto no desempenho de cada técnica e o papel da técnica NVIDIA DLSS como componente viabilizador para as mais avançadas.

Os resultados demonstraram que, embora a rasterização tradicional ainda se apresente como uma solução sólida para obter taxas de quadros mais altas, a tecnologia DLSS se firmou como uma ferramenta necessária para abrandar o alto custo de desempenho do *Ray Tracing*. A análise mostrou que a combinação de *Ray Tracing* com DLSS no ambiente de teste permitiu uma *gameplay* satisfatória, superando os 30 FPS e oferecendo uma qualidade visual mais realista nos quesitos de reflexos, sombras e iluminação se comparada a rasterização.

No entanto, o mesmo não foi possível com *Path Tracing*. Apesar dos ganhos na qualidade visual inquestionável e da fidelidade sem precedentes das imagens, a tecnologia se mostrou impraticável no *hardware* de testes mesmo com auxílio do DLSS. Seu desempenho no ambiente de testes foi drasticamente reduzido a taxas de quadro que comprometeram severamente a jogabilidade, permanecendo muito abaixo dos 30 FPS aceitáveis. A análise reforçou a indicação de que o *Path Tracing* opera em taxas de quadro extremamente baixas. Conclui-se, dado isso, que o custo do fotorrealismo apresenta grandes impedimentos para um hardware de entrada. Enquanto o DLSS propicia uma experiência satisfatória na adoção do *Ray Tracing*, o custo para adotar o *Path Tracing* é extremamente alto, tornando a tecnologia restrita a hardwares de alta performance.

Referências

- Corporation, N. (2018). What's the Difference Between Ray Tracing and Rasterization? <https://blogs.nvidia.com/blog/whats-difference-between-ray-tracing-rasterization/>.
- Corporation, N. (2025a). DLSS 4: Transforming Real-Time Graphics with AI. <https://research.nvidia.com/labs/adlr/DLSS4/>. Technical Report.

- Corporation, N. (2025b). NVIDIA DLAA (Deep Learning Anti-Aliasing). <https://www.nvidia.com/en-us/geforce/news/nvidia-rtx-games-engines-apps/>.
- Corporation, N. (n.d.). NVIDIA DLSS (Deep Learning Super Sampling). <https://developer.nvidia.com/rtx/dlss>.
- Hu, D., Zhang, Z., Hou, T., Liu, T., Fu, H., and Gong, M. (2023). Multiscale representation for real-time anti-aliasing neural rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17772–17783.
- Kelling, J. (2024). *Memory-efficient Real-time Path Tracing for Computer Games*. PhD thesis.
- Kelling, J., Ströter, D., and Kuijper, A. (2024). Light path guided culling for hybrid real-time path tracing. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 7(3):1–17.
- Lenovo (2024). What is anti-aliasing.
- Li, J., Chen, Z., Wu, X., Wang, L., Wang, B., and Zhang, L. (2024). Neural super-resolution for real-time rendering with radiance demodulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4357–4367.
- Mohanto, B., Kluge, S., and Staadt, O. (2024). Towards accelerating real-time path tracing with foveated framework. *arXiv preprint arXiv:2406.07981*.
- Neri, F. V. V. (2022). O que é Path Tracing e qual a diferença do Ray Tracing? <https://www.tecmundo.com.br/software/248569-path-tracing-diferenca-ray-tracing.htm>.
- NVIDIA (2020). What is ai upscaling.
- Spjut, J., Madhusudan, A., Watson, B., Boudaoud, B., and Kim, J. (2022). The esports frontier: Rendering for competitive games. *arXiv preprint arXiv:2208.11774*.
- StudioBinder (2021). What is motion blur: Definition and examples.
- Tan, Y. W., Cui, X., and Bhojan, A. (2022). Hybrid mblur: A systematic approach to augment rasterization with ray tracing for rendering motion blur in games. *arXiv preprint arXiv:2210.06159*.
- Tan, Y. W., Low, S. E., Chow, J., Teo, J., and Bhojan, A. (2024). Dhr+ s: distributed hybrid rendering with realistic real-time shadows for interactive thin client metaverse and game applications. *The Visual Computer*, 40(7):4981–4991.
- Vries, T. d. (2023). *Implementing real-time ray tracing in Unity to increase the render quality of a ray tracing visualization tool*. PhD thesis.
- Wen, Y., Zhang, C., Xie, C., and Fu, X. (2025). Achieving lightweight super-resolution for real-time computer graphics. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 8313–8322.